Student Name   :   JAYANTH P
Roll No   :   7376221CS177
Seat No   :   239
Project Id   :   39
Project Title   :   BIT Staff Quarters Portal

Technical Components:

Components	Tech Stack
Front-end	React JS
Back-end	Spring Boot
Database	MySQL
API	Restful services
Domain	Quarters

Implementation Timeline

Phase	Deadline	Status	Notes
Stage 1	03/05/2024	Approved ▾	Planning and Requirements Gathering
Stage 2		In Progress ▾	Design and Prototyping
Stage 3		Not Started ▾	DB Designing
Stage 4		Not Started ▾	Backend Implementation
Stage 5		Not Started ▾	Integration and Testing
Stage 6		Not Started ▾	Deployment

PROJECT-FLOW:

Purpose:

The purpose of the customized portal for BIT staff quarters is to streamline and centralize the management of staff quarters and facilitate efficient communication and coordination among the residents, administration, and other stakeholders. It aims to provide a user-friendly platform for managing inmate and family member details, check-in and check-out processes, complaint registration, garden cleaning schedules, and the tracking of faculties in and out during general permission (GP) sessions.

Scope:

The scope of the portal includes the development of a web-based platform that allows authorized users to input, view, and manage inmate and family member details, submit and track check-in and check-out forms, register and monitor

complaints, schedule and manage garden cleaning activities, plumbing activities, electrical and record the in and out movements of faculties during GP sessions.

Business Context:

The customized portal addresses the need for a digital solution to manage the staff quarters efficiently. It aims to enhance communication, transparency, and accountability within the community, and streamline administrative processes for better operational efficiency. By providing a centralized platform for various activities and information, the portal seeks to improve the overall living experience within the BIT staff quarters.

Considerations:

1. Data security and privacy of the residents and staff members

2. User-friendly interface for easy access and navigation

3. Integration with existing systems and databases

4. Scalability to accommodate future expansion and additional features

5. Accessibility for different user roles and permissions

Dependencies:

1. Availability of necessary resources for development and implementation

2. Integration with the existing infrastructure and systems

3· Compliance with relevant regulations and policies

User Personas:

1. Resident Inmates: Staff members residing in the quarters.

2. Family Members: Dependents and family members of the staff residing in the quarters.

3. Family Members Details: Bio-Data => Name, age, gender, mailID, quarters, occupation, blood-group, dob, mobile, aadhar, address.

4. Administrative Staff: Personnel responsible for managing the staff quarters, including check-in/out processes, complaint handling, and general operations.
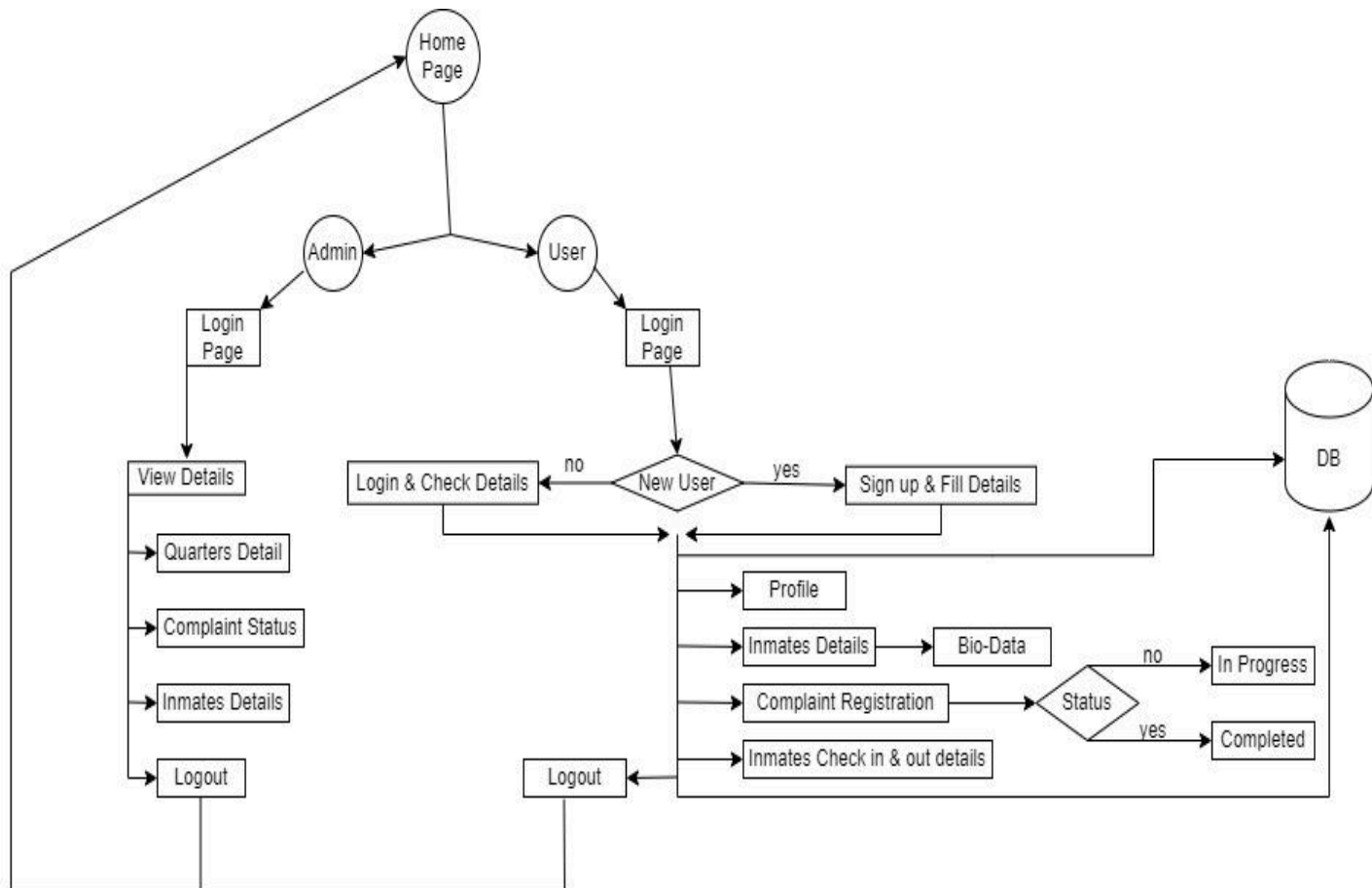
User Stories:

1. As a resident inmate, I want to easily register my family members' details and update them as necessary.

2. As a family member, I want to be able to check in and check out of the staff quarters conveniently.

3. As an administrative staff member, I want to be able to efficiently handle and track complaints raised by the residents.

4. As a resident inmate, I want to be able to easily schedule garden cleaning activities and track their completion.

5. As an administrative staff member, I want to be able to monitor the in and out movements of faculties during GP sessions for security and administrative purposes.

Functional Requirements:

1. User authentication and role-based access control

2. Data entry and management for inmate and family member details

3. Check-in and check-out forms with notification features

4. Complaint registration and tracking system

5. Garden cleaning schedule and tracking functionality

6. Recording and monitoring of faculties in and out movements during GP
    sessions

Flow Chart:

ER Diagram:

Staff Quarters Management System

facultiesInOuts ⤴
id	int
facultyId	int
inTime	timestamp
outTime	timestamp

faculties 💼
id	int
name	string

checkIns →]
id	int
checkInTime	timestamp
inmateId	int

checkOuts [→
id	int
checkOutTime	timestamp
inmateId	int

complaints ⊙
:d	int
inmateId	int
description	string

inmates 👤
.d	int
name	string
familyId	int

families 👥
id	int
address	string
totalmembers	int

gardenCleanings 🔧
id	int
inmateId	int
date	timestamp