

# Mini Project-I (K24MCA18P)

## Odd Semester

## Session 2024-25

**Smart Utility Management System**

**Our Team Member**

**Deepu Kumari ,Dhwani Panchal,Anshu patel**

**Project Supervisor:**

**SupervisorName :-  
ms.divya singhal**

# Content

- Introduction (1 slide)
- Literature Review (2 slides)
- Objective of the Project (1 slide)
- Technology
  - Hardware Requirements (Development Environment, Server requirement (if required), Client requirement (if required)).
  - Software Requirements (Language and Platforms like Frameworks, VS code, Android Studio and Jupyter notebook etc. )
- Modules (2-3 slides)
- Workflow (1 slide)
- Reports (For Example: Project : Student Monitoring System, so reports like: Student Marks, Subjects, companies visit, and student appears in placement etc.)
- References (1 slide)

# Introduction

## **Title: Smart Utility Management System**

### **Overview:**

This project aims to create a web-based booking system for managing essential services (like plumbers, electricians, Cleaning and other utility services) within a society. The system allows residents to raise service requests, schedule bookings, track status, and leave feedback.

# Literature Review

## **Existing Systems:**

- Traditional booking systems like phone-based or in-person requests.
- Service aggregators like UrbanClap, but not tailored to small residential societies.

## **Challenges:**

- Lack of centralized management in societies.
- Poor scheduling and tracking of services.

# Literature Review (Contd.)

## **Need for Real-time Status Updates:**

- Users must stay informed about the status of their service requests (Pending, In Progress, Completed).

## **Handling Booking Conflicts Efficiently:**

- If a service takes longer than scheduled, the system should automatically notify and reschedule affected bookings.
- Ensures fair allocation of services and avoids customer frustration.

## **Rating and Feedback Mechanism:**

- Users can leave detailed feedback and rate service providers based on performance, punctuality, and behavior.
- This promotes accountability and ensures quality service through transparent feedback loops.

# Objective of the Project

## **Primary Objective:**

To develop a centralized utility service management system that simplifies booking, tracking, and feedback within a society.

## **Goals:**

- Enable easy service booking and management.
- Implement efficient scheduling with conflict resolution.
- Ensure transparency with feedback and ratings.

# Technology Stack

## **Frontend:**

- HTML, CSS, JavaScript

## **Backend:**

- Node.js, Express.js
- MongoDB Atlas (Cloud Database)

## **APIs & Tools:**

- Nodemon (development environment)
- CORS (for API security)
- Firebase (Authentication)

# Technology (Software Requirements)

## **Development Tools:**

- **Code Editor:** VS Code
- **Database:** MongoDB Atlas
- **Package Manager:** npm
- **Version Control:** GitHub (optional for team collaboration)



# Technology (Hardware Requirements)

## **Development Environment:**

- A system with at least 8 GB RAM, i5/i7 processor, stable internet connection
- OS: Windows/Linux/macOS

## **Server Requirements (If Hosted Online):**

- Cloud Hosting
- Node.js environment with MongoDB support

## **Client Requirements:**

- Any modern web browser (Chrome, Firefox, Edge, Safari)

# Modules

## **Module 1: User Management**

- User registration and login system (with roles: Admin, Resident)
- Password management

## **Module 2: Service Booking Management**

- Raise a service request (plumber, electrician, etc.)
- Scheduling (start/end time with conflict handling)
- Real-time booking status tracking

## **Module 3: Feedback and Rating**

- Option to leave feedback after service completion
- View aggregated service ratings

# Workflow/Gantt Chart

- **User Registration** → User creates an account with personal details.
- **Login & Dashboard** → Users log in to view available services and bookings.
- **Booking Service** → Select service type, choose time, and confirm the booking.
- **Service Status Tracking** → View status updates (Pending, In Progress, Completed).
- **Feedback Submission** → After service completion, submit feedback and ratings.

# Reports

## **Service Reports:**

- Total services requested (e.g., plumber, electrician, etc.).
- Most frequently requested service types.
- Service completion times and delays.

## **User Reports:**

- Number of active residents vs. inactive ones
- User feedback summary (average ratings).

## **Scheduling Reports:**

- Conflicts resolved and rescheduled bookings.

# References

## **Web Technologies:**

- MDN Web Docs for HTML, CSS, and JavaScript.
- Node.js Official Documentation.
- MongoDB Atlas Documentation.

## **Online Learning Platforms:**

- FreeCodeCamp, Udemy (for backend and frontend development).