

## Project Report: Household Services Application - V2

Author:

Podey Om Prabhakar

21f2000968

[21f2000968@ds.study.iitm.ac.in](mailto:21f2000968@ds.study.iitm.ac.in)

An active passionate coder wearing a smile and motivation over face all time.

### Project Details:

The project aims to develop a comprehensive platform for home servicing solutions with a focus on service management, user roles, and efficient backend operations. The platform supports three types of users: Admin, Service Professionals, and Customers. The Admin manages service professionals, creates and manages services, and monitors customer interactions. Service professionals accept or reject service requests and provide services based on customer requirements. Customers can browse, request, and review services.

### Approach to the Problem Statement:

To develop the Household Services Application (V2), I focused on creating a scalable platform for three main roles: Admin, Service Professionals, and Customers. The system was designed with a **Role-Based Access Control (RBAC)** model to manage permissions for each user type.

Key features include:

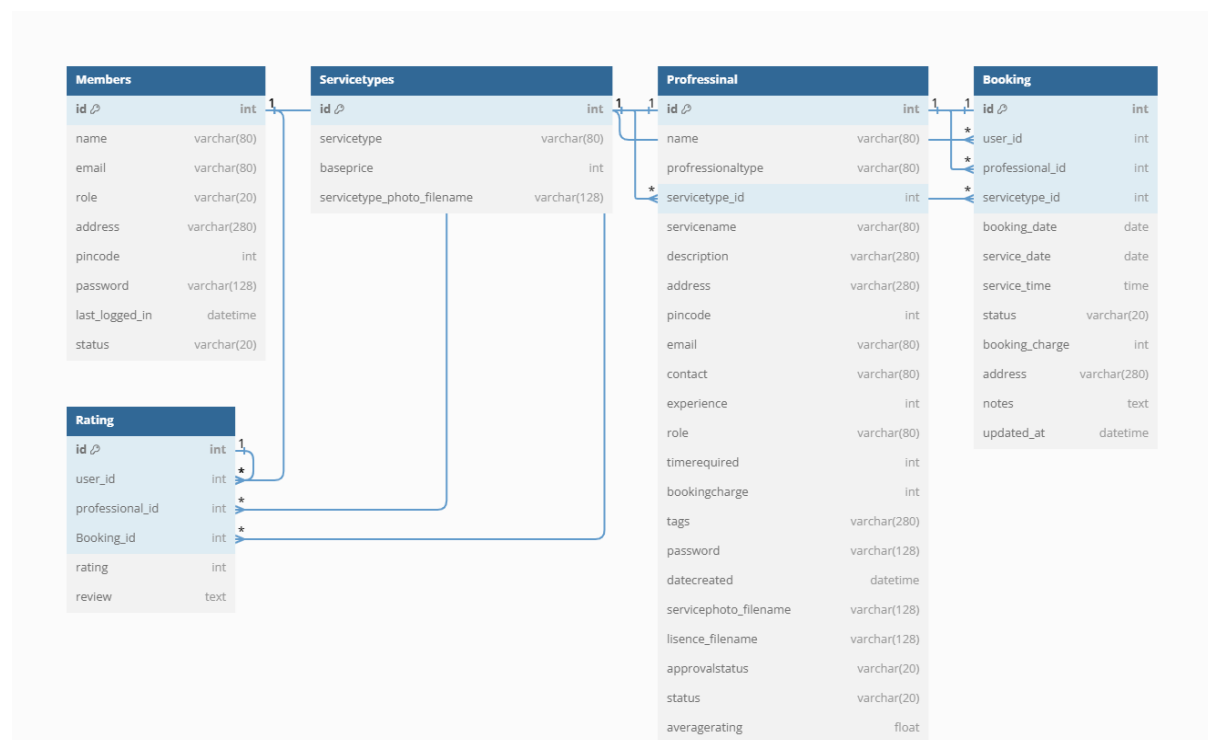
- **Admin Functionality:** Admins can manage services, approve/block professionals, and monitor requests.
- **Service Requests:** Customers can create, edit, and track service requests, while professionals can accept/reject them.
- **Search Functionality:** Customers can search for services by location and type.
- **Backend Jobs:** Implemented **Celery** for tasks like daily reminders and monthly reports.
- **Database Design:** Used **SQLite** to store data, ensuring proper relationships between users, services, and requests.
- **Frontend Development:** Built with **VueJS** and **Bootstrap** to ensure a responsive, mobile-first UI.

Performance was optimized using **Redis caching** for faster data retrieval, and **Celery** for handling batch jobs asynchronously. The project was thoroughly tested to ensure a smooth and reliable user experience.

## Frameworks and Libraries Used:

- **SQLite:** Used for data storage and management.
- **Flask:** Provides the backend API for the application.
- **VueJS:** Used for frontend UI development.
- **Redis:** Implemented for caching to improve performance.
- **Celery:** Used for batch jobs, including daily reminders and monthly reports.

## ER Diagram:



Presentation Video link :

<https://drive.google.com/file/d/1gWQchm7PHJpxGBKN7ccGclioplGfuFEH/view?usp=sharing>

Thank You.