Household Services Application

Project Report by Venkatesh A M - 23F3004331

Modern Application Development – I
September, 2024 Term

Student Details

- **Name**: Venkatesh.A.M
- **Roll No.**: 23F3004331
- **Email**: 23F3004331@ds.study.iitm.ac.in
- **About Me**: I am enthusiastic about building applications that simplify everyday tasks for users. This project allowed me to deepen my skills in backend development, specifically with Flask and database management using SQLite.

Project Overview

The **Household Services Application** aims to streamline the process of connecting customers with service professionals for a variety of household needs, such as plumbing, electrical repairs, and AC maintenance. The platform features three distinct user roles:

- **Customers**: Can register, log in, and request various household services.
- **Service Professionals**: Manage service requests assigned to them and update service statuses.
- **Admin**: Oversees user activities, manages service requests, and ensures proper operation of the system.

Project Approach

The project was developed in a systematic and incremental manner. The steps followed include:

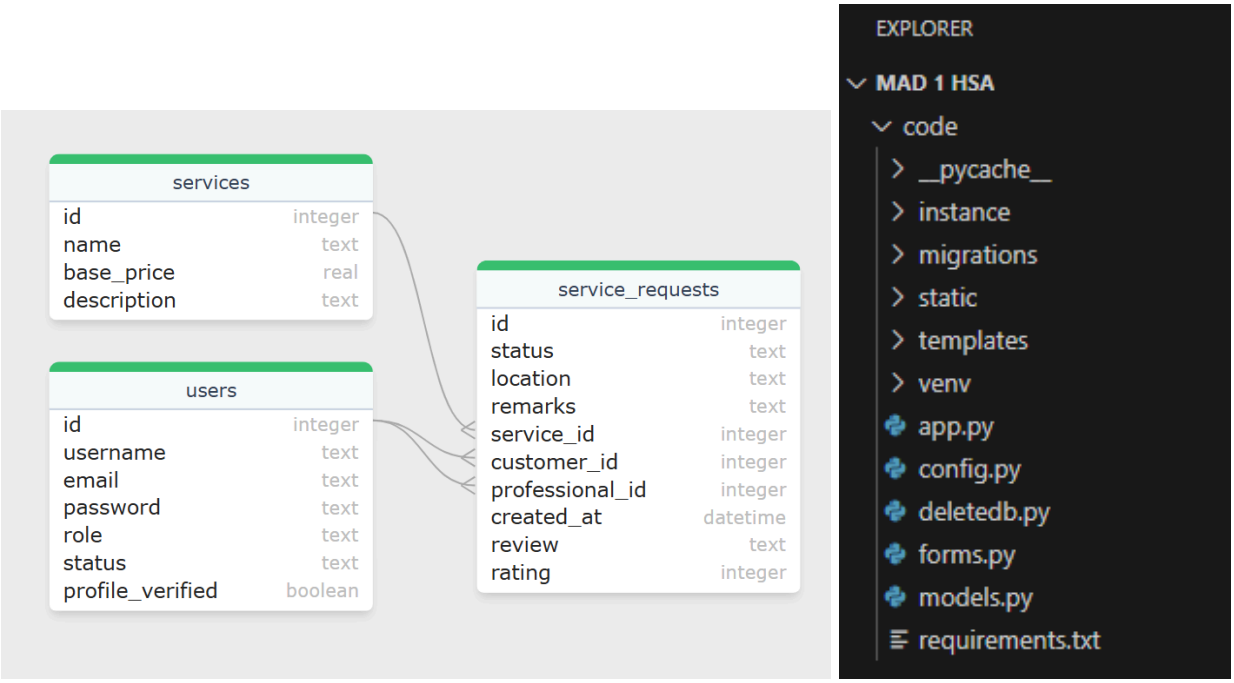
1. **User Role Design**: Clearly defining the functionalities of customers, service professionals, and admins, ensuring a coherent user journey for each role.
2. **Backend Development**:Implemented core functionalities such as authentication, service request creation, and role-based access control using Flask.
3. **Frontend Design**:Incorporated Bootstrap for styling, emphasizing vibrant colors with a dark theme for a modern look.
4. **Testing and Debugging**:Performed rigorous testing of features like service request workflows, user interactions, and role-specific dashboards.

Frameworks and libraries used : Flask, SQLite, Jinja2, Flask-Login, Bootstrap

Database Schema Design

The database is designed with relational tables to handle data effectively across different user roles. Key tables include:

- **Users**: Contains `id`, `username`, `password`, and `role` to distinguish between admins, professionals, and customers.
- **Services**: Details each available service, including fields like `service_id`, `name`, `payment`, and description.
- **Service Requests**: Tracks customer requests with attributes such as `request_id`, `service_id`, `customer_id`, `professional_id`, `status`, and optional `remarks`.
- **Professionals**: Holds information about each professional, including their `id`, `expertise`, and `profile_verified` status.



Project Video

23f3004331_MAD1_projectvideo.mp4
https://drive.google.com/file/d/18Q6SNa8OYoORR17uXiMwDLwbtU3OMwE1/view?usp=sharing

Testing and Future Improvements

The app was tested locally to validate functionality across various user roles. For future development, additional improvements such as data analytics or service feedback tracking could enhance the platform's utility and user engagement.