Household Services Application

Final Project Report

BY ASIF SHARAFAT ANSARI
21F3003001 |Modern Application Development - I | Sep, 2024 Term

AUTHOR

Name – Asif Sharafat Ansari
Roll No – 21f3003001
Email – 21f3003001@ds.study.iitm.ac.in
About me - I combine a background in Electronics and Telecommunication with a passion for coding to create impactful applications and solutions. Proficient in Python, web development frameworks like Flask, and cloud computing, I excel at building user-focused projects. I also enjoy sharing knowledge, learning new technologies, and collaborating on innovative ideas.

DESCRIPTION OF PROJECT

The Household Services Application is a multi-user platform designed to connect customers, service professionals, and administrators seamlessly. It provides a one-stop solution for various home services, including cleaning, plumbing, electrical repairs, and more. Customers can easily browse services, request bookings, and track service history, while professionals manage their offerings and availability. Admins oversee platform activity, ensuring smooth operations and resolving disputes. This comprehensive app simplifies home maintenance, delivering convenience and reliability to all users.

TECHNOLOGIES USED

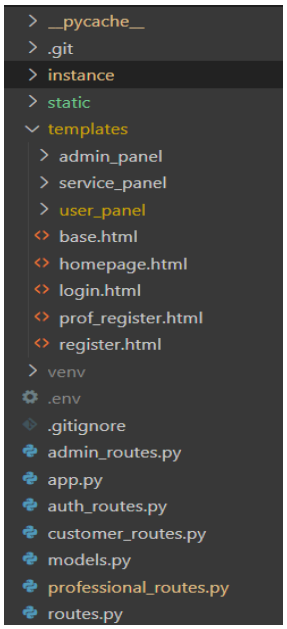
- **Flask**: Backend framework for building the web application.
- **SQLAlchemy**: ORM (Object-Relational Mapping) tool for database interactions.
- **SQLite** : Database management system for storing application data.
- **HTML/CSS/JavaScript/Bootstrap**: Frontend technologies for user interface design and interactivity.
- **Flask-Login**: Extension for managing user sessions and authentication.
- **Datetime**: Python library for handling date and time operations.
- **werkzeug.security**: This module provides utilities for securely handling passwords and other sensitive data.

ARCHITECTURE

Here, the app.py file contains the main code to run the Household Services web app. HouseholdServices is a module that contains all the files for the application. The description of the files inside HouseholdServices are as follows:

- **App.py** – This file serves as the entry point for a Flask application, managing route definitions, application configuration, and server execution.
- **Routes.py** – This file contains backend logic like login, registration

- **Admin_Routes.py** – This file contains admin dashboard backend logic like adding Services, Professionals, Users with their functionalities
- **Customer_Routes.py** – This file contains users backend logic like booking Services, providing feedback along with other functionalities
- **Professional_Routes.py** – This file contains service professionals backend logic like Accepting/Rejecting Services booked by users, along with other functionalities
- **Models.py** – This contains DB Classes
- **Auth_Routes.py** – This contains login/logout logic handling
- **Templates** – This contains all HTML templates for Admin, Users, Professionals
- **Static** – This contains CSS and image files



FEATURES

User Management

- Registration and authentication for users (Admin, Service Professionals, and Customers).
- User roles to control access and permissions for each user type.
- User profile management, including personal details like name, email, contact number, and address.

Service Management

- Creation and management of services with attributes like name, base price, time required, and description.
- Classification of services into categories for easy navigation and discovery.
- Admin can update or delete services as required.

Service Request Management

- Customers can create new service requests based on available services.
- Tracking service request details, including status (requested, assigned, closed) and remarks.
- Professionals can view, accept/reject, and complete assigned service requests.

Search and Filter

- Customers can search for services by name, location, or pin code.
- Admins can search for professionals to block/unblock or review their profiles.

Professional Management

- Admin approval process for professionals after verifying profile documents.
- Professional profile management, including name, description, service type, experience, and ratings.
- View customer reviews for professionals to maintain quality assurance.

Admin Dashboard

- Monitor all users (customers and service professionals) and their activities.
- Block users (customers or professionals) based on fraudulent activity or poor reviews.
- Oversee and take actions on pending service requests.

Customer Dashboard

- View available services with details like description, price, and time required.
- Track service history, including service details, status, and professional assigned.
- Post reviews or remarks for completed services.

Feedback System

- Collection of customer reviews and ratings for completed services.
- Display reviews to other customers for transparency.

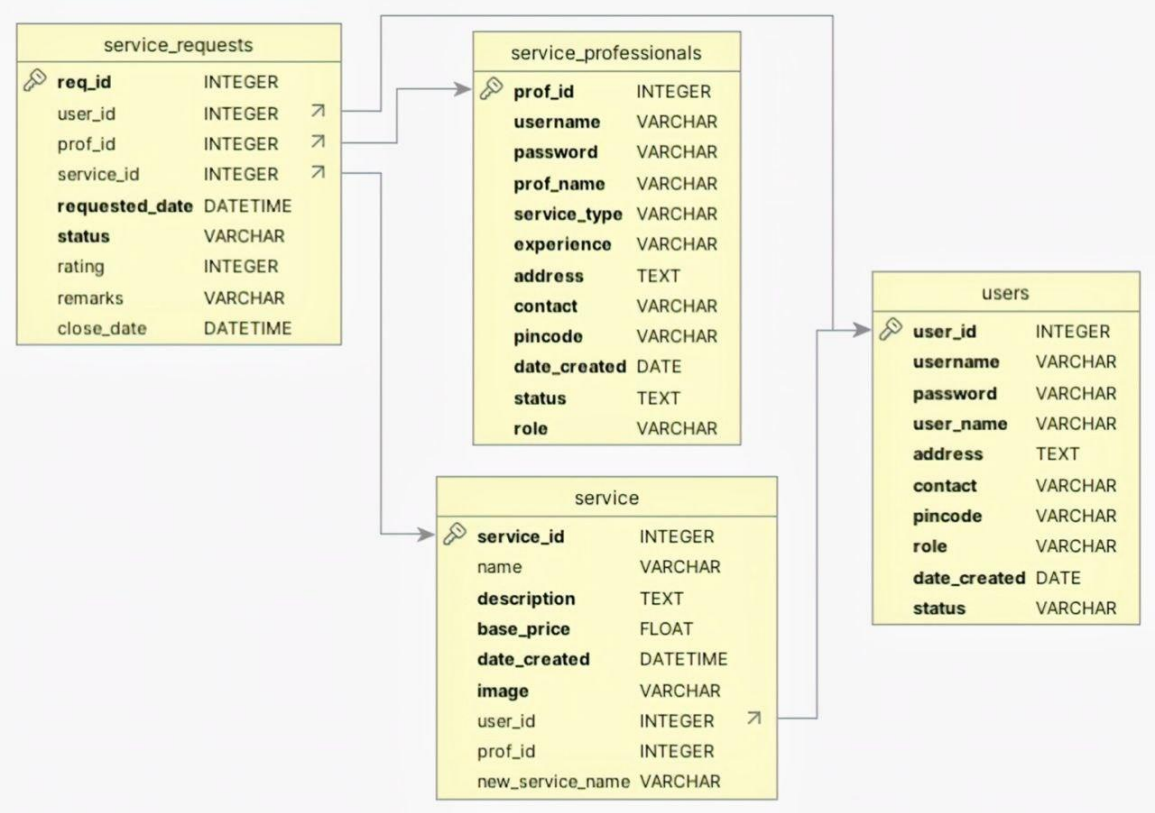
Notifications and Updates

- Notify professionals about new service requests.
- Send reminders to customers for pending actions like service completion or reviews.

DB SCHEMA DESIGN

The database schema for the Household Services Application includes tables for users (Admin, Service Professionals, and Customers), services, service requests, bookings, feedback, and professional profiles. Relationships are established between these tables to track user activities such as service requests, professional assignments, and feedback

submissions. Each table contains relevant fields such as user details (name, email, role), service information (name, price, description), request statuses (requested, assigned, completed), book/close dates, and customer reviews, ensuring comprehensive data management within the application.



FUTURE ENHANCEMENTS

- Implementing a reservation system to allow customers to book services at specific times.
- Enhancing the user interface for a more intuitive and seamless experience.
- Integrating email notifications for service request updates, status changes, and reminders.
- Implementing graphs and data visualizations to track service trends and professional performance.
- Allowing secure online payments and adding a review system for customer feedback.

VIDEO LINK

Video demonstration of my project is available here.

https://drive.google.com/drive/folders/1xD7_NT9VEqr5Z3fPo6S0NAEvc-cLwP4A?usp=drive_link

THANK YOU