

Household Services app

(Modern Application Development – I)

Student details

Name – Ustela Sukesh Reddy

Roll no. – 23f2000504

Email – 23f2000504@ds.study.iitm.ac.in

Project Overview

Qwix.it serves as a platform for connecting service professionals with customers seeking various household services. The application will enable customers to easily book services, track their requests, and provide feedback. Service professionals can accept or reject service requests, manage their requests and packages. The admin will oversee the platform, manage user accounts, and ensure smooth operations. .

My Approach

1. Database Schema Design:

Designed the database schema, which serves as the foundation for storing and managing the application's data, and its relationships between tables for clarity.

2. User Flow and Wireframing

To visualize the user journey and interaction with the application, I created detailed wireframes, outlining the layout, navigation, and UI elements for each screen.

3. Template Creation:

I constructed the HTML templates using Jinja2, according to the wireframe designs. Each template was designed to be visually appealing, enhancing user engagement.

4. Flask Route Implementation:

I defined Flask routes to handle different user actions and requests. These routes were organized based on user roles (admin, service professional, and customer). Then I implemented CRUD operations for various entities, such as users, services, and service requests. Finally additional features, such as search functionality, review systems, and notifications, were integrated.

Additional Features Implemented

Review Section: A section to view all reviews on professional dash, and on service packages for customers

Saved Packages: Option to save desired packages and view all saved packages on customer dash

Flag Packages: Ability for an admin to flag a suspicious package before deleting it

Special Services: Platinum (premium) and emergency service sections

Technologies used

Backend:

Flask: Python micro web framework

SQLAlchemy: Object-Relational Mapper (ORM)

SQLite: Database management system

Frontend:

HTML: Structure of web pages

CSS: Styling of web pages

Bootstrap: CSS framework (used for accordion element) **Chart.js:** JavaScript library for creating charts and graphs

Additional Libraries and Tools:

Flask-Session: Session management for storing user data across requests

Datetime: Date and time handling

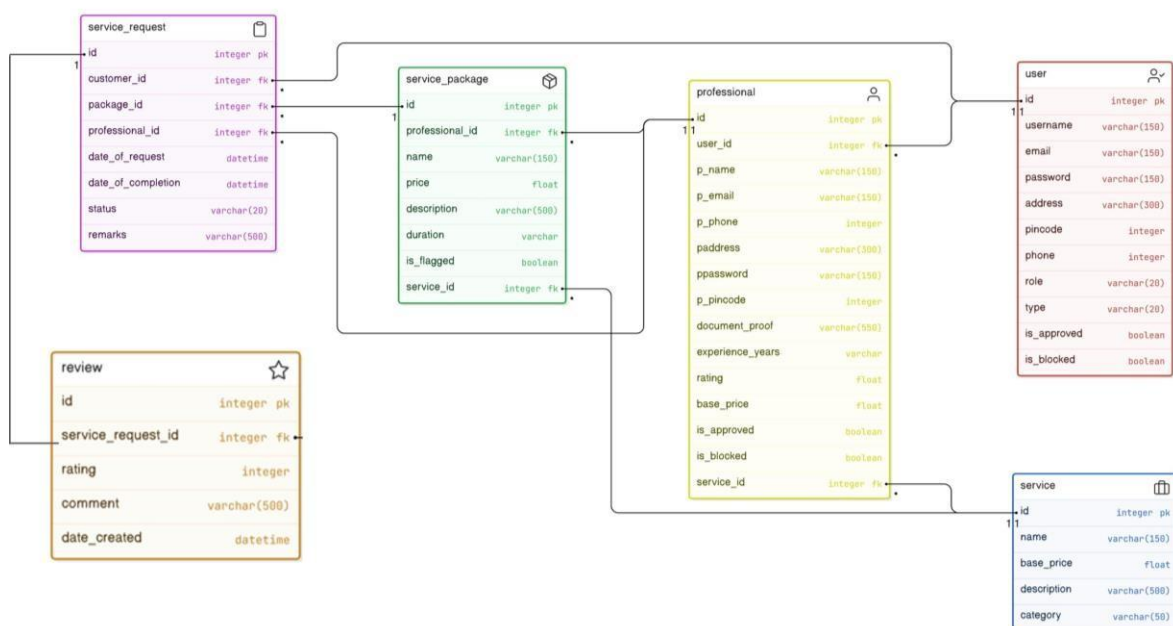
Werkzeug: Secure password hashing and authentication

Pytz: Library for working with time zones (timezone)

Sqllalchemy.func: Provides functions for performing database operations like aggregation

Sqllalchemy.or_: Function for building OR clauses in SQL queries.

E-R Diagram



Wireframe

Project Structure

Demonstration Video

https://drive.google.com/file/d/19OuRvR0WfIVpOI-lvzbIM5nMOzq_bm5K/view?usp=sharing