J company

(House Hold Services Application)

Student Details

- **Name:** Jatish Chawla
- **Roll Number:** 23f2000832
- **Course/Branch:** BS in Data Science

Problem Statement

Develop a multi-user platform to connect customers with verified service professionals for household services such as plumbing, electrical work, and cleaning. The platform enables:

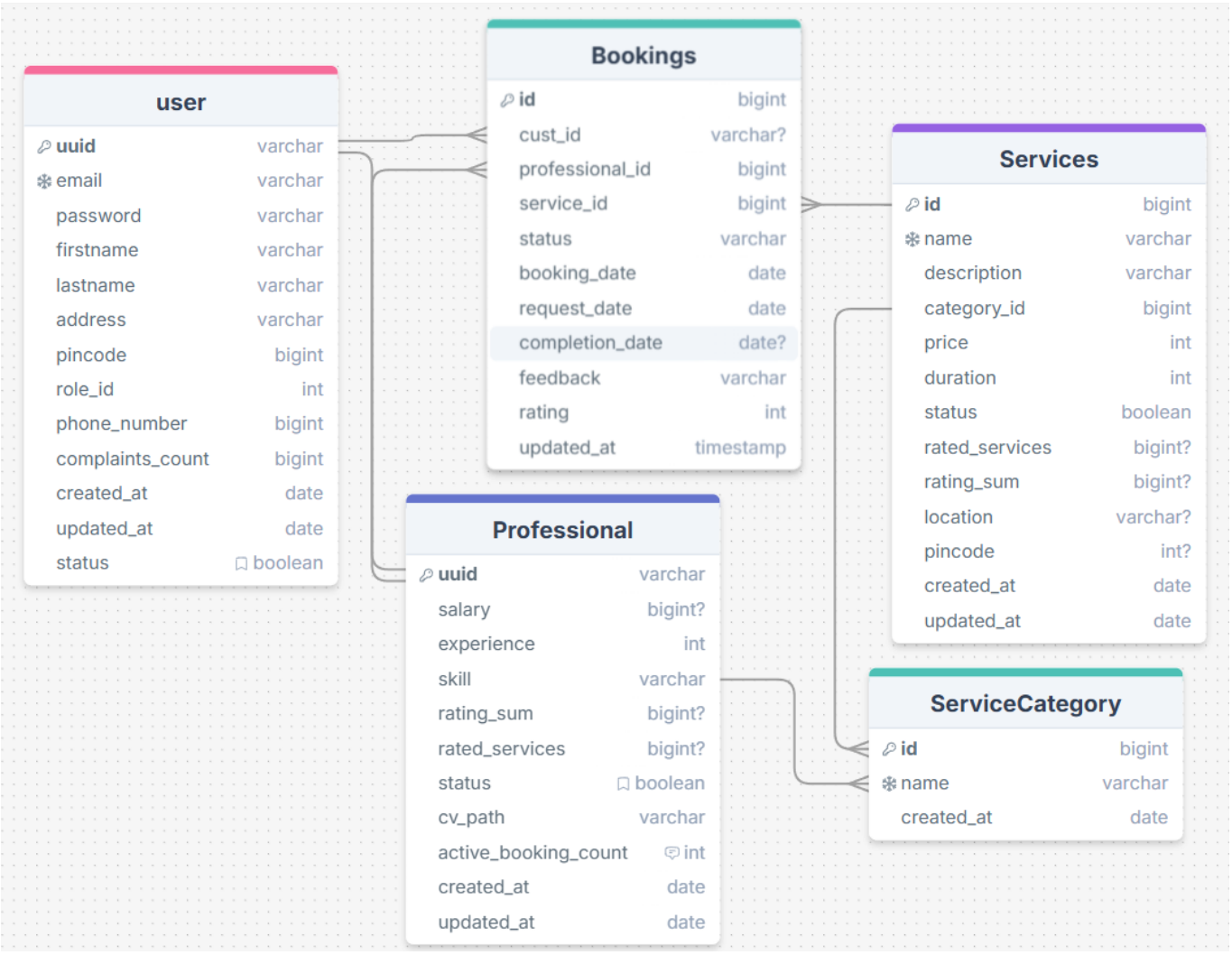
- **Customers** to search and filter services, create and manage service requests, and provide feedback post-completion.
- **Service Professionals** to accept or reject requests, complete tasks, and build a reputation through customer reviews.
- **Admins** to oversee the system, manage users, services and categories, approve service professional profiles, and address fraud or performance issues.

The platform ensures streamlined booking, accountability, and reliable services for an enhanced user experience.

Approach to the Problem Statement

1. **Problem Understanding**:
   The primary goal was to create a system that bridges the gap between customers in need of household services and professionals offering these services. The application needed to streamline booking, communication, and management for both customers and professionals.
2. **Requirement Analysis**:
   - Identified three user roles: **Admin**, **Customer**, and **Service Professional**.
   - Admin should manage services, users, and monitor bookings.
   - Customers should be able to browse services, book them, and view their booking history.
   - Professionals should manage their profiles, upload CVs, and accept/reject service requests.
3. **Designing the Database**:
   - Created a relational database with normalized (3NF) tables for **Users**, **Services**, **Categories**, **Professionals**, and **Bookings**.
   - Ensured proper relationships between tables (e.g., a booking links customer, professional, and service).

ER Diagram:



Frontend and Backend Design:

The project uses Python (Flask) for backend logic, database interactions, and routing. The frontend uses HTML, CSS,and Bootstrap 5.3.0 for responsive and visually appealing designs, integrated with Flask templates for forms, modals, and dashboards.

Libraries and Frameworks Used

- **Flask**: Backend framework.
- **Flask-SQLAlchemy**: Database ORM.
- **Flask-Login**: User authentication.
- **Werkzeug** : secure file handling
- **Jinja2**: Templating engine.
- **SQLite**: Database.

Admin Capabilities in the Application:

1. **Service Management**
   - Add, update, or delete services with details like price, description, and duration.
2. **User Management**
   - Manage customer and professional accounts and can view, edit and delete profiles.
   - Block or unblock users based on ratings, reports or feedback.
3. **Request Oversight**
   - View all service requests and their statuses.
   - Monitor service request trends and completion rates.
4. **Analytics Dashboard**
   - Access dynamic charts showing statistics service requests.
   - Can view total services, customers, professionals and total revenue
5. **Feedback Review**
   - View customer ratings and feedback on services.
6. **System Maintenance**
   - Ensure platform integrity by overseeing key database entries and operational data.

Key Features

1. **Service Management**: Admins manage services (add, update, delete) and can overall ratings.
2. **Service Booking Requests**: Customers create, view, edit, and manage requests.
3. **Professional Profiles**: Professionals manage profiles, accept/reject requests.
4. **User Authentication**: Secure login/registration for all users. Used hashing(bcrypt) to store passwords.
5. **RBAC**: Implemented Role Based Access Control
6. **Feedback System**: Customers rate and review services. Helps in maintaining quality.
7. **Report system**: Dedicated Report system that allow customers to report a professional and professional to report a customer. This helps in tracking fraudulent activities.
8. **Dynamic Charts**: Dynamic charts provide a clear and interactive representation of the platform's current statistics. These charts are implemented across admin, customer, and professional dashboards. Real-time stats on dashboards (requests, ratings, feedback), Customers can see their summary charts for bookings, professionals can get their analytical information in form of chart.
9. This streamlined platform ensures efficient service management, user-friendly interactions, and insightful analytics.
10. **Search and filter:**
    - **Customers:** customers can search for available services, filter by service name, category, location, pin code, price, duration and rating
    - **Service Professional:** can search for the service history either canceled or closed, filter by Service name, Date, Address, Pin code.
    - **Admin:** Admin can search for Services, Categories, Users (customers, Service Professionals), Bookings,

API Resource Endpoint: /admin/api/schema

- **Description**: This API provides the database schema information, including table names, columns, data types, and metadata such as nullability and primary keys.
- **Method**: GET
- **Authorization**: Only accessible by users with the role id of 1 (Admin). Unauthorized users will be redirected and shown an error message.
- **Response**: A JSON object containing the database schema details.

DRIVE LINK OF THE PRESENTATION VIDEO:

https://drive.google.com/file/d/1WBqtE8y2t47j45fVMvgjiyafgznXN9XK/view?usp=drive_link