Project On “Modern Application Development I Course”

TITLE: A-Z HOUSEHOLD SERVICES

AUTHOR:
Name: KHUSHI TIWARI
Roll No: 22f2000364
Email: 22f2000364@ds.study.iitm.ac.in
AI ,ML AND WEB DEV ENTHUSIAST || DATA DRIVEN INSIGHTS ADVOCATE

1.Description:

Objective was to build a web application for household services platform focusing on using Flask as the backend framework for the application. Having the additional feature of providing admin/user-based facilities on the website. In this project, we were supposed to use HTML, CSS, bootstrap, flask, SQL Alchemy, and other necessary modules to build an app for coordination platform. We had to build a login/signup page where we can store the usernames and passwords of people who have visited this web app. Here, users can create profile and perform the provided required operations. Profiles can be created, deleted, and updated. User can track their progress of by looking at the details presented in the dashboard.

Note:

To run the application, install flask, flask SQLAlchemy. And run the “app.py” file using the “python” command on the terminal.

2.Technologies used:

i. **Flask:** used for building the web application.
ii. **Flask-SQLAlchemy:** extension of Flask, used to handle database connections across the app.
iii. **CSS:** CSS is employed for styling the web pages and enhancing the user experience.
iv. **HTML**: HTML is used for structuring the web pages and creating user interfaces.
v. **ChartJs**: Uses for creating different types of charts on admin dashboard
vi. **SQL lite**: Database management system for storing application data.
vii. **Jinja2**: Template engine for rendering dynamic HTML content.

3.DB Schema Design:

Entities and Attributes:

1. User: User ID (primary key), username, password, role, and status

2. CustomerInformation: User ID (foreign key referencing User), email, address, and Pin code.

3. ProfessionalInformation: User ID (foreign key referencing User), email, service name, and experience.

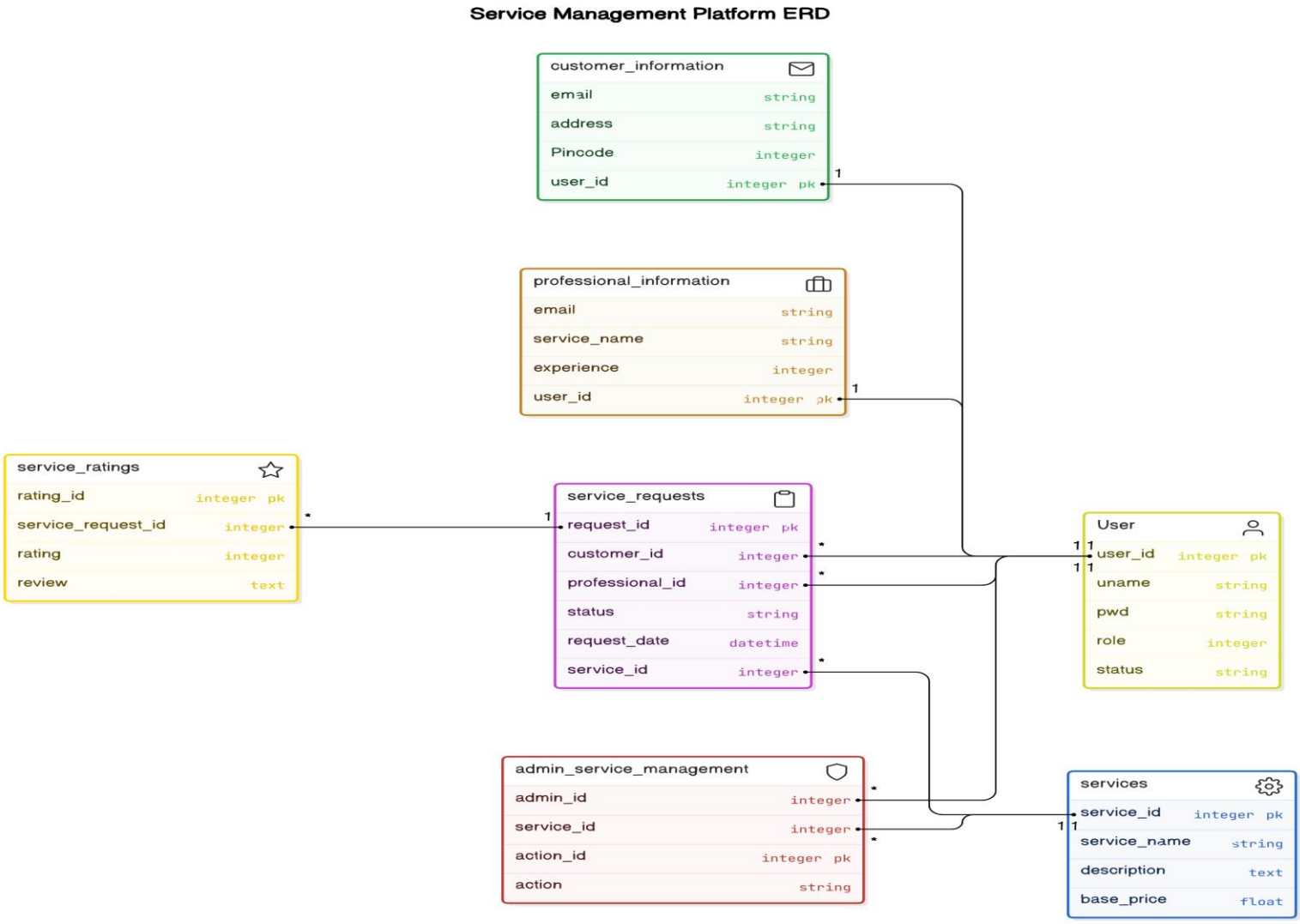
4. Service: Service ID (primary key), service name, description, and base price.

5. ServiceRequest: Request ID (primary key), customer ID (foreign key), professional ID (foreign key), service ID, status, and request date.

6. ServiceRating: Rating ID (primary key), service request ID, rating, and review.

7. AdminServiceManagement: Action ID (primary key), admin ID, service ID, and action.

ER Diagram:



5.Features:

1. The **flask app** opens at the main page. Where person can

i. User login: Username and Password are required fields in form. User should exist in the database or he/she can register himself/herself.
ii. Admin login: Username and Password are required fields in form. Admin cannot register, id is created at the start itself.
iii. Customer Signup: Name, Email, Address, Password and Pin code are required in form.
iv. Professional Register: Name, Email, Service Name, Password, Experience are required in form.

2. Users

a. Customer: The customer can send request to book services to the professional. Customer can also search the services provided by the professionals under search functionality. Customer can rate the Professional also, based on the service provided.

b. Professional: The professional can either accept or reject the requests made by the customer. The professional can also see the requests that were closed by the customer.

c. Admin**:**

- Can login, and see all the statistics in dashboard.
- An admin can monitor all the service requests.
- Admin can edit/delete the services and also approve/reject/delete the professional's register request.

6. VIDEO:

HTTPS://DRIVE.GOOGLE.COM/FILE/D/1--8-3JFAK8KPM6MSDBAKRJ5EINWIKYZO/VIEW?USP=SHARING