Author

Ayush Singh
21f3001194
21f3001194@ds.study.iitm.ac.in
I am a passionate student of data analysis and web development pursuing dual degree in Computer Applications and Data Science, and I am a resident of Lucknow.

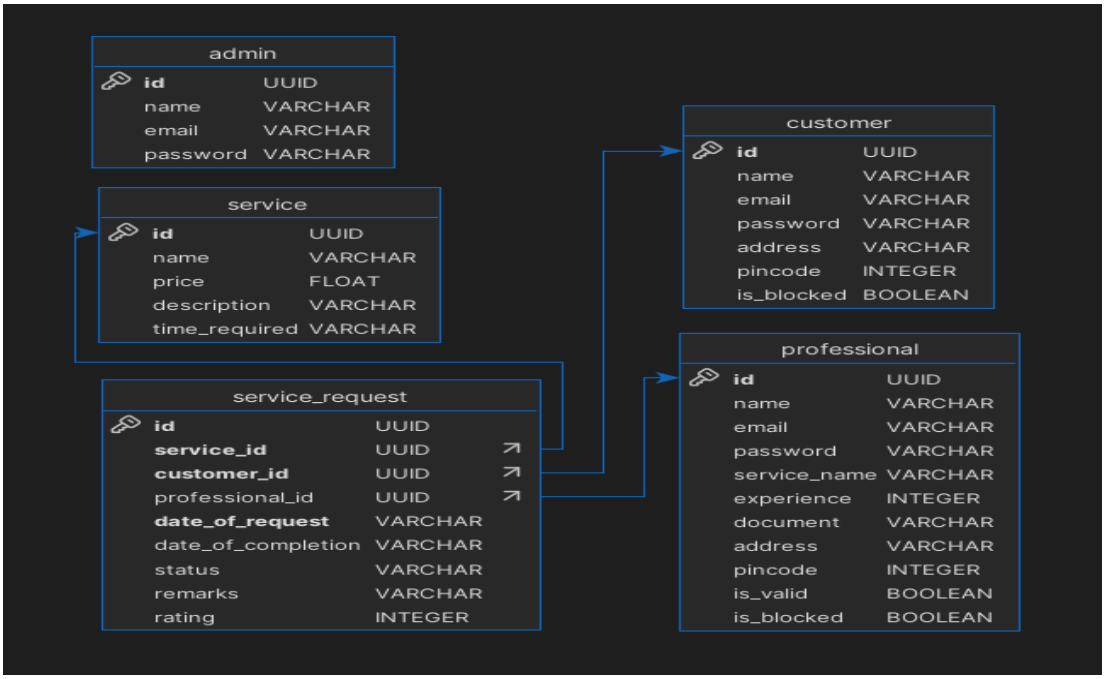
Description

Create a software system which recruits workers for a service, which are controlled by the Admin. Customers make requests to make use of the worker's profession and pay them for It. Use a valid framework to make it systematic and in working condition.

Technologies used

- **Flask** for application code
- **Jinja2** templates + **Bootstrap** for HTML generation and styling
- **SQLite** for data storage
- **Flask extensions**: flask-sqlalchemy for database connectivity, flask-login for managing logins, flask forms, and Werkzeug library.

DB Schema Design



I have formulated **5 tables** in the database, one for the **admin** only, the rest are for **services**, **customer**, **professionals** and **service requests**(links services, professionals & customer).

Architecture and Features

The project is organised inside the root folder consisting of the following folders:

- **Instance** folder to contain database file
- **Static** folder containing images
- **Templates** folder containing role specific sub-folders and templates.
- The **Uploads** folder contains verification documents uploaded by professionals.
- 3 python files as: **app.py, forms.py, and models.py** containing project code, schema, routes and forms.

The features implemented in the project are as follows:

- **Multi-role login** and registration, and user specific dashboards.
- **Admin dashboard**, admin can monitor customers, professionals, create new services, approve service professionals after verification, and block fraudulent people.
- **Customers** can make service requests based on the services available, edit existing services and also close existing services by rating professionals. Customers also can search for services and can get insights from the summary section.
- **Professionals** can accept/reject service requests made by customers and also search for requests. He can see his ratings and request's status in the summary section as well, and check his performance.

Video

2024-11-22 23-30-58.mkv

Conclusion

The project successfully developed a platform that connects household service providers with customers. It is robust, user-friendly, and scalable, with the potential for future enhancements like online payment integration and real-time service tracking.