



IITM

MAD – I Project

----- A to Z Service Management App -----

Created By:

Name :- Daiwik Rankawat

Roll Number :- 22F3000099

Email: 22F3000099@ds.study.iitm.ac.in

Description :

This is a project to build Service Management App using knowledge gained and learned in Modern App Dev 1, The name of App is “ A to Z Service Management App ”.

Technology used:

1. **SQLite – Python (sqlite3 and flask):** The main Technologies SQLite Flask by Python. I choose sqlite3 instead of SQL – alchemy due to bind data end to end points.
Libraries that are used in this project are :-
 - **Sqlite3** : Used for connecting SQLite Database.
 - **Flask**: A lite, open – source web framework which is Python based it has some libraries:
 - **Flask**: It is used for handle HTTP request (like restful Apis like GET, POST, etc.)
 - **render_template**: It is used for rendering Jinja2 templates (html pages).
 - **Request**: It is used to capture information from User side in form of data, query parameters, etc.)
 - **Redirect**: It send a response to browser to redirect to different url
 - **url_for**: it allows to route to a specific URL by referring to a function name.
 - **Send_file** : It allows to send a file from the backend server to client page as a download able file (like jpg, pdf, png etc).
 - **send_from_directory**: It will allows a backend file to visualise in frontend (like jpg, png).
2. **Jinja2 and Bootstrap**: Jinja2 was used for templating the UI and bootstrap is used for better CSS and some in-built (Bootstrap JS) JavaScript functions like Dropdown, and etc.

Functionalities :

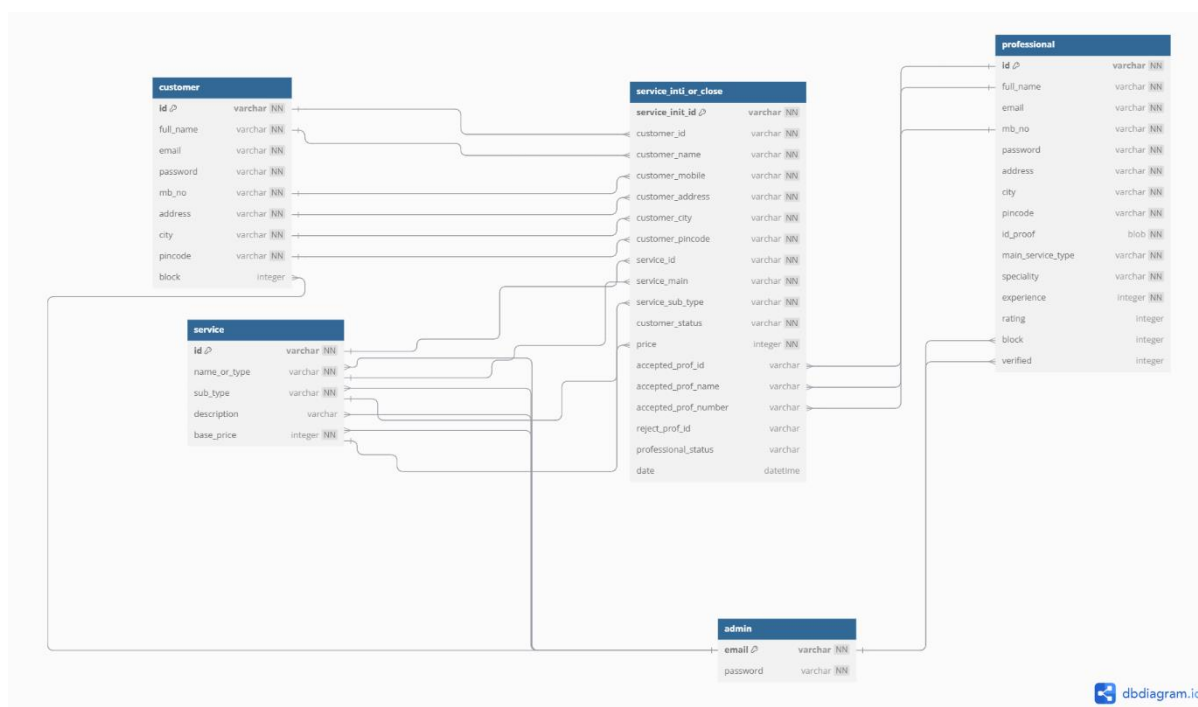
1. Registration : Separate registration page for Customer, Admin, Professionals.
2. Login: Separate login page for Customer, Admin, Professionals.
3. Each type of user has their own UUID (universally unique id).

4. Three different Dashboard Pages for Customer, Professionals, Admins. Each page has some features.
5. Without having UUID no one is able to access their dashboard.
6. Customer are able to create a new service request and also able to check its status.
7. Professionals are able to see available and **Accept** or **Reject** service request of the service type for which they register themselves only Ex: If a service professional is register for haircut in salon type then he/she only can able to see the available service only of haircut service created by customers.
8. All information taken from client side will send to backend server. Some info is send through Forms and some info is send in form of parameters.
9. Admin will able to create a different type of Services while admin is also able to specify the service as well as able to set a price for that particular service.
10. Admin is able to all info related a newly created service request.
11. Admin, Customer, professionals are able to search with some parameters.
12. Search Page is common for Customers, Admins, Professionals Every type of user have different parameter to search.
13. Professionals are able to see there profile own info.

**** Without admin verification professionals are not able to see the service request created by customer**

Database Schema

All Database schema is given shows in pictorial form.



* In service_inti_close table the customer status changes accordingly.

** In service_inti_close table the Professional status changes either None or Accepted.

Complete Info and Video Tutorial Can Be found Here:

[YouTube Link – \(https://youtu.be/yrqrI5WAHhI\)](https://youtu.be/yrqrI5WAHhI) and

Drive Link - <https://drive.google.com/file/d/1PDGp7iQhAE4qT3iNJT9z6j-hupGUaTR9/view?usp=drive>