

MAD-2 PROJECT REPORT

Author Name: Guhan Kannan

Roll No: 23f2001584

Email Id: 23f2001584@ds.study.iitm.ac.in

I am currently admitted in the BS degree Offered by IITM I have completed the foundation level and all theory courses plus MAD-1 project in the Diploma in Programming level currently pursuing Diploma in Data Science and I am also in the second year of my engineering at NIT Thiruchirapalli.

Description:

This project is about a website that acts as an interface between service professionals and customers where there is also an Admin who creates services and customers are empowered to give service requests to professionals, professionals have the ability to act upon it.

Technologies used:

Flask: This is a framework in python used for creating web applications

SQLAlchemy: This is used for database ORM(Object Relational Mapping)

SQLite: It is used to store the database

HTML, CSS – These are used for Markup and Styling respectively.

Bootstrap- This is used to develop web applications that are responsive.

VueJs- This is a JavaScript framework that is used to code the frontend.

JWT-This is used for handling role based authentication.

Redis-This is used for caching

Celery- This is used for Backend jobs.

Database Design:

users :

- user_id (Integer, Primary Key)
- name (String, 80)
- user_name (String, 80)
- email_id (String, 120, Unique)

- password (String, 200)
- role (String, 50)
- is_blocked (Boolean, Default=False)

admins :

- user_id (Integer, Primary Key, Foreign Key to users.user_id)
- admin_id (Integer, Unique)

customers:

- customer_id (Integer, Primary Key, Unique)
- user_id (Integer, Foreign Key to users.user_id)

professionals:

- professional_id (Integer, Primary Key)
- approved (Boolean, Default=False)
- user_id (Integer, Foreign Key to users.user_id)

services:

- service_id (Integer, Primary Key)
- price (Float)
- name (String, 80)
- location (String, 200)
- description (String, 500)

service_requests:

- servicerequest_id (Integer, Primary Key)
- service_id (Integer, Foreign Key to services.service_id)
- professional_id (Integer, Foreign Key to professionals.professional_id)
- customer_id (Integer, Foreign Key to customers.customer_id)
- request_date (DateTime)
- completion_date (DateTime, Nullable)
- status (String, 50)

APIs:

/register – For new user to register.

/login – To login into the web application

/admin/view_users-for admin to view list of users

/admin/approve_professionals- for the admin to approve professionals

/admin/users – for the admin to manage users

/admin/block_users- for the admin to block users

/admin/create_service – for the admin to create services

/admin/manage_services – for the admin to manage the created services.

/customer/view_services-for customer to view list of services

/customer/create_service_request- for the customer to give a service request based on the services to a professional.

/customer/manage_service_requests-for the customer to manage created service requests

/professional/view_services-for professional to view services

/professional/service_requests-for the service professionals to view all service requests

/professional/service_requests/update_status- for the service professionals to update the status of service requests

/customer/search_services-for customer to search for service

/admin/search_professionals-for admin to search for professional

/download-csv-for downloading service request statistics(downloading csv file)

Video:

drive.google.com/file/d/1XKFoGfyxN3HpbbgfECSruFP8PBk0xysR/view