Household Services Application - V2

Project Report

1. Author Details

- **Name: Manish jat**
- **Roll Number: 23f3004152**
- **Email:23f3004152@ds.study.iitm.ac.in**

---

2. Project Overview

The "Household Services Application - V2" is a multi-user web platform connecting customers with service professionals for household services like plumbing and electrical repairs. It provides role-based access for admins, service professionals, and customers, ensuring efficient service management through authentication and automated notifications.

---

3. Technologies Used

- **Backend: Flask (Flask-RESTful, Flask-JWT-Extended) for API development.**
- **Frontend: VueJS for a responsive UI.**
- **Database: SQLite for structured data storage.**
- **Performance: Redis for caching and Celery for background tasks.**
- **Security: JWT-based authentication for role management.**

---

4. DB Schema Design

- **Users Table: Stores user details (ID, Name, Role, Email, Password Hash, Status).**
- **Services Table: Lists services (ID, Name, Base Price, Description).**

- Requests Table: Tracks service requests (ID, Customer ID, Service ID, Status, Created Date).
- Reviews Table: Stores customer feedback (ID, Request ID, Rating, Comment).

The relational database design ensures efficient querying and data integrity through constraints such as foreign keys and indexing.

5. API Design

- `POST /auth/login` - User login.
- `GET /services` - List available services.
- `POST /requests` - Customer creates a request.
- `PUT /requests/{id}` - Update request status.
- `GET /users` - Admin views all users.

The API follows RESTful principles, ensuring scalability and maintainability. A separate YAML file documents the API specifications.

6. Architecture and Features

Architecture:

- Backend: Organized into controllers (API endpoints), models (database schema), and services (business logic).
- Frontend: Vue components for modular and reusable UI elements.
- Asynchronous Tasks: Celery handles time-intensive operations like report generation and notifications.

Key Features:

- Admin Dashboard: User and service management.
- Customer Portal: Service search, request tracking, reviews.
- Service Professional Panel: Request management, status updates.
- Automated Notifications: Email or Google Chat webhooks for reminders.
- CSV Export: Admin can export closed service requests.
- PDF Reports: Monthly reports for customers.
- ChartJS Integration: Visual analytics for service trends.

---

7. Performance Enhancements

- Database Indexing: Faster query execution.
- Caching: Redis for optimized API performance.
- Async Tasks: Celery for scheduled jobs like reminders and reports.

---

8. Video Presentation

🎬 mad2_presentation.mp4

https://drive.google.com/file/d/1ByxPKmiH98BqPqiBAiTITaBpmNYIo-O2/view?usp=drive_link

---

9. Conclusion

This project successfully implements a scalable and optimized household service platform. It integrates structured workflows, authentication, and background task automation, ensuring an efficient and user-friendly experience.

End of Report