Taxi Service Application

GitHub Repository: https://github.com/AveryKillian99/342-Group6/tree/final

Table of Contents

1. Introduction
    a. Project Overview
    b. Objectives
2. Use Cases
    a. User Use Cases
    b. Admin Use Cases
    c. Provider Use Cases
3. Design Document
    a. System Architecture
    b. Technologies Used
    c. Feature Overview
4. Scenarios With Screenshots
5. Group Members and Responsibilities
6. Conclusion
    a. Final Thoughts
    b. Future Enhancements

Introduction

Project Overview

The Taxi Service Application is designed to connect users, providers, and admins. Users can request rides, providers can offer rides, and admins can manage users and services.

Objectives

- Develop a web-based system with role-based access for users, admins, and providers.
- Implement CRUD functionalities for each actor:
    - **User**: Ride requests
    - **Admin**: User management, ride monitoring
    - **Provider**: Ride acceptance

Use Cases

User Use Cases

1. **Register as a User**: The user enters their details to create an account.
2. **Request a Ride**: After logging in, users can request a ride, selecting the pickup location and destination.

Admin Use Cases

1. **Manage Users**: Admins can view, edit, and delete user accounts.
2. **View Ride Requests**: Admins can monitor active and past ride requests.

Provider Use Cases

1. **Accept Ride Requests**: Providers can view ride requests and choose to accept or reject them.
2. **View Ride History**: Providers can see a list of completed rides and their details.

Design Document

System Architecture

The application follows the **MVC (Model-View-Controller)** pattern:

- **Model**: Represents data structures (User, Provider, Admin, Ride).
- **View**: The UI elements for each actor (User, Admin, Provider).

- **Controller**: Handles requests, processes data, and updates views.

Technologies Used

- **Backend**: Java, Spring Boot
- **Database**: MySQL
- **Frontend**: HTML, CSS, JavaScript

Feature Overview

- User registration, login, and ride request functionality.
- Admin dashboard for managing users and ride requests.
- Provider interface for managing and accepting ride requests.

Scenarios With Screenshots

(To be completed with screenshots and descriptions for User, Admin, and Provider scenarios)

Group Members and Responsibilities

- **Avery Killian (User Use Cases)**:

Implemented the user registration process and ride request functionalities.

- **Gia Nguyen (Admin Use Cases)**:

Developed the admin dashboard and user management features.

- **Carter Gibb (Provider Use Cases)**:

Created the provider interface for accepting and managing ride requests.

Conclusion

Final Thoughts

This project demonstrates a functional taxi service application, showcasing the ability to implement role-based functionalities with Spring Boot and MySQL.

Future Enhancements

- Payment integration for processing ride fees.
- Real-time tracking of rides using maps.