

Secure Software Development and Engineering (CY321)



Project Proposal

**Title: Simple Ride-Hailing Website with Integrated
Security Features**

Team Members:

1. Arsalan Khan (2022115)
2. Saad Ali(2022512)

Introduction:

The goal of this project is to develop a simple ride-hailing website that allows users to book rides and communicate with drivers. The website will focus on core functionalities such as user authentication, ride booking, and a chat system between drivers and customers. Payment processing will be outsourced to simplify the project and reduce security risks associated with handling sensitive financial data.

This project will emphasize secure software development practices, including secure authentication, data encryption, and protection against common vulnerabilities such as SQL injection and cross-site scripting (XSS).

Objectives

- Develop a user-friendly ride-hailing website with simplistic features.
- Implement secure authentication and authorization mechanisms.
- Provide a simple chat system for communication between drivers and customers.
- Ensure the website is secure against common vulnerabilities.
- Demonstrate secure coding practices and adherence to security requirements.

Features The website will include the following features:

User Side:

- Secure registration and login.
- Ride booking (pickup and drop-off locations).
- Ride history.
- In-app chat with the driver.

Driver Side:

- Secure registration and login.

- Accept or reject ride requests.
- View assigned rides.
- In-app chat with the customer.

Admin Side (Optional):

- Manage users and drivers.
- Monitor ride requests.
-

Security Requirements The following security measures will be implemented:

Secure Authentication:

- Use bcrypt for password hashing.
- Implement JSON Web Tokens (JWT) for session management.

Data Encryption:

- Encrypt sensitive data (e.g., passwords) in the database.

Access Control:

- Implement role-based access control (e.g., riders can only book rides, drivers can only accept rides).

Input Validation and Sanitization:

- Validate and sanitize user inputs to prevent SQL injection and XSS attacks.

Secure Communication:

- Use HTTPS for secure communication between the client and server.

Chat Security:

- Ensure chat messages are transmitted securely.
- Optionally encrypt chat messages in the database.

Technology Stack

- Frontend: HTML, CSS, JavaScript (React.js or Vue.js for interactivity).
- Backend: Python (Django/Flask) or Node.js (Express.js).
- Database: MySQL or PostgreSQL.
- Chat System: WebSockets (e.g., Socket.io) or a simple REST API for messaging.
- Security Libraries: bcrypt, JWT, OWASP libraries.

Outcome By the end of this project, we will deliver a fully functional, secure ride-hailing website with the following:

- A user-friendly interface for riders and drivers.
- A simple chat system for communication.
- Secure authentication and data protection mechanisms.
- A detailed report documenting the threat model, security features, and test results.
- A live demo showcasing the website's functionality and security features.

Conclusion:

This project will demonstrate our ability to develop secure software while focusing on simplicity and core functionalities. By outsourcing payment processing, we can reduce complexity and focus on delivering a secure and user-friendly ride-hailing platform.