Online Ticket Reservation System - Project Documentation

# Project Overview

An online ticket reservation system developed using Spring Boot (Java) and HTML/CSS/JS frontend. Users can register, login, and book tickets for bus, railway, or air transportation. Bookings are stored in a MySQL database and all operations can be tested via Postman or a user-friendly web interface.

# Modules Implemented

# 1. User Module

- Registration: Allows new users to sign up.

- Login: Authenticates users based on credentials.

Entity: User

- id: Long

- username: String

- email: String

- password: String

API Endpoints:

- POST /api/users/register — Register user

- POST /api/users/login — Authenticate user

Frontend:

- register.html — Styled registration form

- login.html — Styled login form (with link to register)

After Registration: Redirects to login.

After Login: Redirects to booking.html

# 2. Booking Module

- Users can book tickets by selecting transport type, from/to locations, travel date, and seat class.

- Bookings are stored in the MySQL database.

Entity: Booking

- id: Long

- type: String (bus, railway, air)

- fromLocation: String

- toLocation: String

- travelDate: String

- seatClass: String (Seater, Sleeper, AC, etc.)

API Endpoints:

- GET /api/bookings — Get all bookings

- GET /api/bookings/{id} — Get booking by ID

- POST /api/bookings — Create booking

- PUT /api/bookings/{id} — Update booking

- DELETE /api/bookings/{id} — Delete booking

Frontend:

- booking.html — Ticket booking form

- view-bookings.html — Fetches and displays all bookings

After Booking: Redirects to view-bookings.html

# Technologies Used

- Backend: Spring Boot

- Frontend: HTML, CSS, JavaScript (Fetch API)

- Database: MySQL

- Testing Tools: Postman

# File Structure

com.example.reservation

├── controller

│ ├── BookingController.java

│ └── UserController.java

├── model

│ ├── Booking.java

│ └── User.java

├── repository

│ ├── BookingRepository.java

│ └── UserRepository.java

├── service

│ ├── BookingService.java

│ └── UserService.java

├── ReservationApplication.java

Frontend files are placed in src/main/resources/static/

static/

├── login.html

├── register.html

├── booking.html

└── view-bookings.html

# Flow

1. Register → 2. Login → 3. Book Ticket → 4. View All Bookings

# Extra Notes

- Static pages serve as UI (not using Thymeleaf or templates)

- Redirections are handled via JavaScript (window.location.href)

- Passwords are stored as plain text (Note: Use encryption for real-world apps)

# Future Improvements

- Add session management (Spring Security)

- Encrypt passwords

- Filter bookings by user

- Improve UI with frameworks (Bootstrap/Tailwind)

- Pagination for booking records

# Snapshots:







