Local Bus Ticket Reservation System

Console Application using Java, JDBC, Hibernate & MySQL

Raghuvaran D
AF04966797
ANP-D1938
ANUDIP FOUNDATION

Introduction

•A bus ticket reservation system designed to handle ticket booking operations.
•Built using **Java**, **JDBC**, **Hibernate**, and **MySQL**.

Helps users:
•Add buses & passengers
•Book tickets
•View records
•Update & Delete records

Problem Statement

Traditional ticket booking involves manual processes that are:

Problem	Impact
Manual booking	Time-consuming
Error-prone records	Inaccurate passenger data
No central database	Hard to track bookings
Difficult record retrieval	Delays & customer dissatisfaction

Objectives

✓ Automate the ticket booking process
✓ Store data securely using Database
✓ Provide CRUD operations:

Create, Read, Update, Delete

✓ Maintain Bus, Passenger & Ticket records
✓ Enable future upgrade to Spring Boot web application

System Architecture

- User → Java Application → Hibernate ORM → MySQL Database

Layers Used:

- Entity Layer (Bus, Passenger, Ticket)
- DAO Layer (Data Access)
- Service/Controller Layer (Business Logic)
- Database Layer (MySQL)

Technologies Used

Technology	Purpose
Java	Core development
JDBC & Hibernate	ORM & DB connectivity
MySQL	Database
Eclipse/IntelliJ	IDE & execution
Maven	Build management
(Future) Spring Boot	Web application development

Key Features

- **Current Features**
- Add Bus, Passenger, and Book Tickets
- View all records
- Search (Bus/Passenger/Ticket)
- Update & Delete Records

🌐 **Future Scope with Spring Boot**

- Convert to Online Web-based Portal
- Add Login Authentication
- Payment Gateway
- Email/SMS Ticket Alerts
- REST API for mobile app

Sample Database Schema (MySQL)

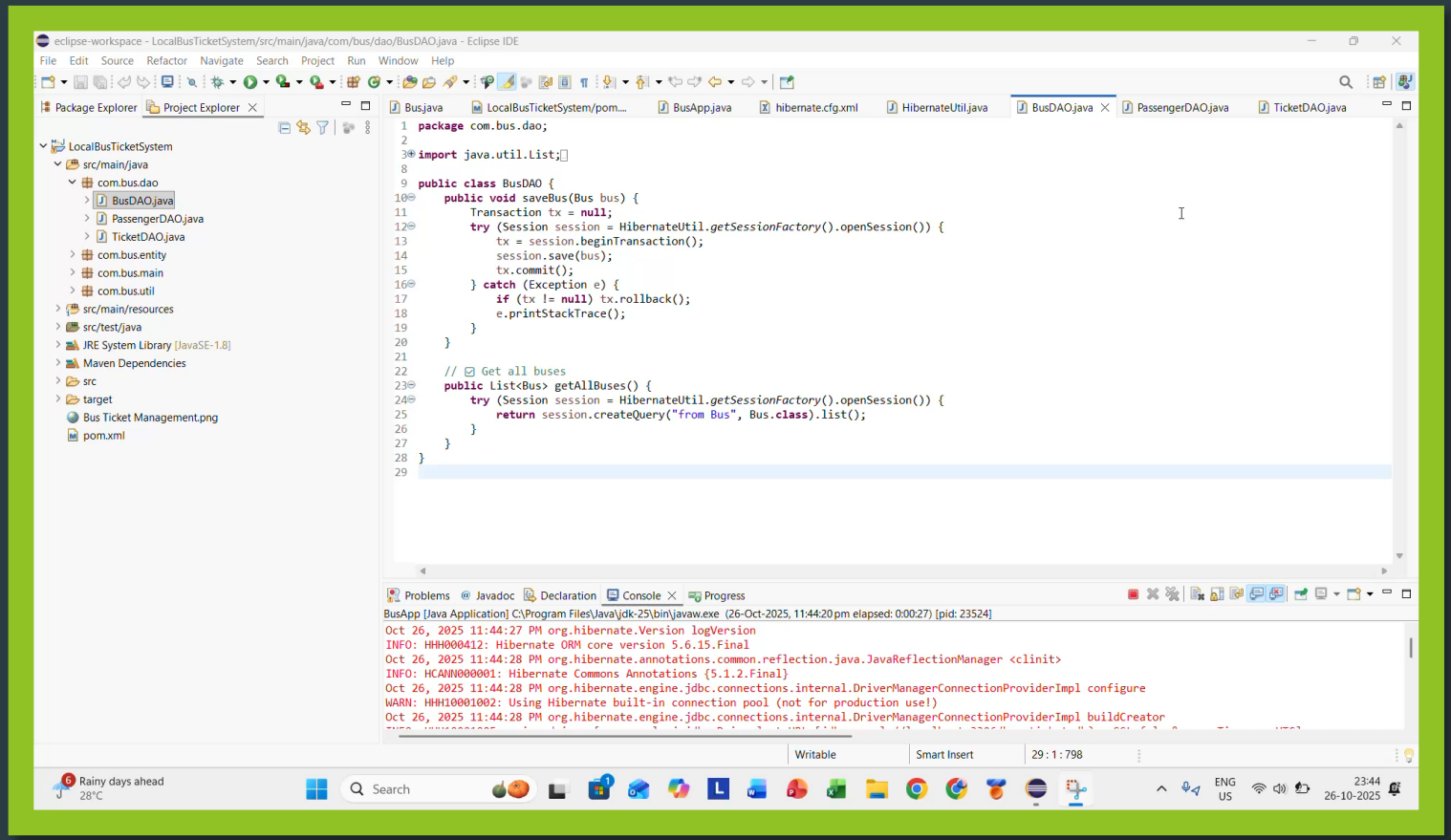
Tables Used:

- bus(id, busName, route, totalSeats)
- passenger(id, name, age)
- ticket(id, bus_id, passenger_id, date, seatNo)

Relationships:

- One Bus → Many Tickets
- One Passenger → Many Tickets

Results


eclipse-workspace - LocalBusTicketSystem/src/main/java/com/bus/dao/BusDAO.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Project Explorer
LocalBusTicketSystem
src/main/java
com.bus.dao
BusDAO.java
PassengerDAO.java
TicketDAO.java
com.bus.entity
com.bus.main
com.bus.util
src/main/resources
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
src
target
Bus Ticket Management.png
pom.xml
Bus.java
LocalBusTicketSystem/pom....
BusApp.java
hibernate.cfg.xml
HibernateUtil.java
BusDAO.java
PassengerDAO.java
TicketDAO.java
package com.bus.dao;
import java.util.List;
public class BusDAO {
public void saveBus(Bus bus) {
Transaction tx = null;
try (Session session = HibernateUtil.getSessionFactory().openSession()) {
tx = session.beginTransaction();
session.save(bus);
tx.commit();
} catch (Exception e) {
if (tx != null) tx.rollback();
e.printStackTrace();
}
}
// Get all buses
public List<Bus> getAllBuses() {
try (Session session = HibernateUtil.getSessionFactory().openSession()) {
return session.createQuery("from Bus", Bus.class).list();
}
}
}
Problems Javadoc Declaration Console Progress
BusApp [Java Application] C:\Program Files\Java\jdk-25\bin\javaw.exe (26-Oct-2025, 11:44:20 pm elapsed: 0:00:27) [pid: 23524]
Oct 26, 2025 11:44:27 PM org.hibernate.Version logVersion
INFO: HHH000412: Hibernate ORM core version 5.6.15.Final
Oct 26, 2025 11:44:28 PM org.hibernate.annotations.common.reflection.java.JavaReflectionManager <clinit>
INFO: HCANN000001: Hibernate Commons Annotations {5.1.2.Final}
Oct 26, 2025 11:44:28 PM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl configure
WARN: HHH10001002: Using Hibernate built-in connection pool (not for production use!)
Oct 26, 2025 11:44:28 PM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl buildCreator
Writable
Smart Insert
29 : 1 : 798
Rainy days ahead
28°C
Search
ENG
US
23:44
26-10-2025

Conclusion & Future Enhancements

- The system provides a complete ticket reservation flow using Java + Hibernate. Data is stored safely in MySQL with structured architecture.
- **Future Enhancements:**
  - ✅ Spring Boot + Thymeleaf / React UI
  - ✅ Online booking system
  - ✅ Reporting Dashboard
  - ✅ Mobile App Integration

References

- https://hibernate.org/
- https://spring.io/projects/spring-boot
- https://dev.mysql.com/doc/
- https://mvnrepository.com/