

■ Vehicle Registration & Licensing System

Project Overview

A system that manages vehicle registrations, license renewals, fines, and inspection schedules. Users can register vehicles, renew licenses, and check fines. Admins can approve registrations, assign inspectors, and manage vehicle records.

Business Logic Highlights

- 1 Vehicle license expires every 1 year → Renewal required.
- 2 If renewal is late → Apply fine automatically.
- 3 Car inspection must be valid (within 6 months) before license renewal.
- 4 Each vehicle belongs to a specific owner.
- 5 Admin approval is needed for new vehicle registration.

Main Entities

- 1 User
- 2 Vehicle
- 3 License
- 4 Inspection
- 5 Fine
- 6 Admin

Example API Endpoints

Vehicle Management

- 1 POST /vehicles/register – Create new vehicle (pending admin approval).
- 2 GET /vehicles/{id} – Get vehicle details.
- 3 GET /vehicles/owner/{ownerId} – List all vehicles of a user.
- 4 PUT /vehicles/{id}/approve – Admin approves registration.

License Management

- 1 POST /licenses/renew/{vehicleId} – Renew vehicle license.
- 2 GET /licenses/{vehicleId} – Get license info for a vehicle.
- 3 GET /licenses/expired – List expired licenses.

Inspection Management

- 1 POST /inspections/{vehicleId} – Add inspection record.
- 2 GET /inspections/upcoming – Show inspections due soon.

Fine Management

- 1 GET /fines/{ownerId} – Show all fines for an owner.
- 2 POST /fines/pay/{fineId} – Mark fine as paid.

Project Tasks for Junior Developers

Phase 1 – Setup & Entities

- 1 Create Spring Boot project structure.
- 2 Configure database connection (MySQL).
- 3 Create entities: User, Vehicle, License, Inspection, Fine.
- 4 Add relationships (OneToMany, ManyToOne).

Phase 2 – Repositories & Services

- 1 Implement CRUD Repositories.
- 2 Write Service layer for each entity.
- 3 Add simple validation (e.g., plate number unique).

Phase 3 – Vehicle Registration

- 1 API: Register new vehicle.
- 2 Admin endpoint: Approve or reject registration.
- 3 Add status field (PENDING / APPROVED / REJECTED).

Phase 4 – License Renewal Logic

- 1 Check if inspection is valid before renewal.
- 2 Calculate fine if license expired.
- 3 Update license expiry date after renewal.

Phase 5 – Inspections

- 1 Add inspection record API.
- 2 Auto-calculate next inspection due date (6 months later).
- 3 Endpoint to list vehicles needing inspection soon.

Phase 6 – Fines

- 1 Generate fine automatically when renewal is late.
- 2 Add endpoint to view and pay fines.
- 3 Mark fine as paid in database.

Phase 7 – Testing & Documentation

- 1 Use Postman to test all endpoints.
- 2 Write README with API documentation.
- 3 Add example request/response bodies.

Optional Enhancements

- 1 JWT authentication for Users/Admins.
- 2 Email notification when license is near expiry.
- 3 Dashboard endpoint for Admin summary (e.g., number of active licenses, total fines).