Project Title

EV Fleet Telemetry & OTA Update Management System

1. Problem Statement

Modern electric vehicle fleets generate massive real-time telemetry data (battery health, speed, location, temperature).
However, **most academic projects stop at static dashboards** and fail to address:

- Real-time ingestion at scale
- Vehicle health monitoring
- Fault detection
- Safe, controlled firmware updates

There is a strong gap between **theoretical IoT concepts** and **real-world fleet-scale systems** used in industry.

2. Proposed Solution

We propose a **modular EV Fleet Management Platform** that evolves across two semesters:

- **Minor Project:** Build a **robust telemetry ingestion, monitoring, and alerting system**
- **Major Project:** Extend the same system to support **Over-The-Air (OTA) firmware updates**, fleet orchestration, and analytics

3. Minor Project Scope

Core Objectives

Focus only on **telemetry**, not OTA.

Functional Modules

1. **Vehicle Telemetry Simulator**
    - Simulates multiple EVs
    - Generates data: speed, SOC, battery temperature, GPS
    - Configurable transmission rate
2. **Telemetry Ingestion Service**
    - Spring Boot backend
    - Receives telemetry via TCP/WebSocket/REST
    - Validates and processes incoming data
3. **Data Storage**
    - PostgreSQL / Time-series storage
    - Efficient querying for historical trends

4. **Monitoring & Alerts**
    - Rule-based alerts (low SOC, high temperature)
    - Alert logs for fleet operators
5. **Web Dashboard**
    - Live vehicle status
    - Telemetry charts
    - Alert visualization

Outcome of Minor Project

A **working, real-time EV telemetry monitoring platform** with clean architecture and extensibility.

4. Major Project Scope (7th Semester – Expansion Plan)

The **major project builds directly on the minor**, no rewrite.

Added Modules

1. **OTA Update Manager**
    - Firmware versioning
    - Secure update delivery
    - Rollback on failure
2. **Fleet-Wide Update Orchestration**
    - Batch updates
    - Canary deployments
    - Vehicle compatibility checks
3. **Scalable Messaging Layer**
    - Event streaming for telemetry & updates
    - Fault-tolerant processing
4. **Advanced Analytics**
    - Battery degradation trends
    - Predictive fault detection (ML-based)
5. **Security & Access Control**
    - Vehicle authentication
    - Role-based access for fleet admins

Technology Stack

- **Backend:** Spring Boot (Java)
- **Frontend:** React.js
- **Database:** PostgreSQL / Time-series DB
- **Communication:** WebSockets / TCP
- **Simulation:** Python-based vehicle simulator

(Advanced tools like Kafka, Kubernetes introduced only in Major phase)

The minor project focuses strictly on telemetry ingestion and monitoring.
OTA update delivery, large-scale orchestration, and DevOps automation are part of the **major project extension**.