**💼 Project Overview: Employee Management System (EMS)**

The **Employee Management System (EMS)** is a **full-stack enterprise-grade web application** designed for companies to manage their internal workforce operations. It facilitates **employee and manager interaction**, improves operational efficiency, and supports real-time collaboration and monitoring.

**🔧 Technology Stack**

**🖥️ Frontend (React)**

* **React with Vite**
* **Tailwind CSS** for UI styling
* **React Router DOM** for routing
* **React Hook Form + Yup** for form validation
* **Axios** for API handling
* **JWT Authentication** (Protected routes)

**⚙️ Backend (Spring Boot Microservices)**

* Built using **Spring Boot Microservices architecture**
* **Maven** for dependency management
* **Spring Security + JWT** for authentication and authorization
* **Spring Web, JPA, RestTemplate, Feign, Eureka, Gateway** (planned)
* **MySQL** for relational data storage

**🗃️ Database**

* Uses **MySQL** with a normalized schema
* Contains essential entities like:
  + Employee, Manager, Attendance, LeaveRequest, Feedback, Task, Ticket, ChatMessage

**🌐 Architecture Summary**

**🔀 Microservices (Backend)**

We are dividing the backend into the following Spring Boot microservices:

| **Service** | **Responsibility** |
| --- | --- |
| **Auth Service** | Login, JWT token generation & validation |
| **User Service** | Employee and Manager CRUD operations |
| **Attendance Service** | Attendance marking and viewing |
| **Leave Service** | Leave request handling with status filters |
| **Feedback Service** | Daily feedback and replies |
| **Task Service** | Task assignment and status updates |
| **Ticket Service** | Ticket creation and tracking |
| **Chat Service** | Real-time messaging |
| **API Gateway** | Routing to all services |
| **Eureka Server** | Service discovery |

**🔑 Key Features**

**👨‍💼 Manager Panel**

* Login using credentials
* View all employees
* Add new employee
* Manage leaves (Pending, Approved, Declined)
* View and reply to daily feedback
* Manage attendance of employees
* Assign tasks and track progress (Pending, Working, Completed)
* View tickets assigned to employees
* Chat with employees in real time

**👨‍💻 Employee Panel**

* Login using credentials
* Fill daily feedback
* Mark their own attendance
* View assigned tasks
* Request for leave
* View ticket status
* Chat with manager in real time

**🔐 Authentication & Authorization**

* Role-based access control using **JWT**
* Separate dashboards for **Employee** and **Manager**

**📦 GitHub & Workflow Strategy**

* **Two separate repositories**:
  + ems-frontend (React)
  + ems-backend (Spring Boot Microservices)
* **Branching Model**:
  + main → production-ready
  + dev → integration branch
  + Individual branches: harpreet, kk, nikhil (per developer)
* **Backup/testing branch** for safe feature merging

**🎓 Learning Objectives**

This project is also built as a **learning environment** where developers will practice:

* **Spring Core Topics**: Collections, Threads, JWT, Security, REST APIs
* **React Core Topics**: Hooks, Routing, State Management, Axios, Form Handling
* **Microservices & Architecture**
* **GitHub Collaboration & CI/CD**
* **Database Design (ERD, SQL)**