

Project: Employee Performance and Feedback Management System

1. Introduction

This document outlines the Low-Level Design (LLD) for an **Employee Performance and Feedback Management System**, which provides a structured way to evaluate employee performance, collect feedback from managers and peers, and track progress over time.

This design supports both **Java (Spring Boot)** and **.NET (ASP.NET Core)** frameworks for backend development.

2. Module Overview

2.1 Employee Profile Management Module

- Allows the registration and updating of employee profiles with personal, role, and performance data.
- Supports the management of employee details, including department, position, and performance history.

2.2 Performance Review Module

- Enables managers to schedule and conduct regular performance reviews for employees.
- Tracks performance metrics and generates performance reports.

2.3 Feedback Collection Module

- Facilitates the collection of feedback from peers, managers, and direct reports.
- Consolidates feedback to provide a comprehensive view of employee performance.

2.4 Goal Management Module

- Enables employees to set and track progress on professional goals.
- Allows managers to assign goals and monitor their achievement.

2.5 Reporting and Analytics Module

- Generates detailed reports and analytics on employee performance, feedback, and goal achievement.
- Provides insights for HR and management to make data-driven decisions.

3. Architecture Overview

3.1 Architectural Style

- **Frontend:** Angular or React for an interactive user interface.
- **Backend:** REST API-based architecture to handle requests and business logic.
- **Database:** Relational Database (MySQL/PostgreSQL/SQL Server) for storing employee, feedback, and performance data.

3.2 Component Interaction

- Frontend communicates with the backend through REST APIs for managing employee data, reviews, and feedback.
- The backend interacts with the database to store and retrieve employee profiles, reviews, goals, and feedback.

4. Module-Wise Design

4.1 Employee Profile Management Module

4.1.1 Features

- Register new employees and update their profiles.
- Store personal, contact, and professional information such as role and department.

4.1.2 Data Flow

- Employees or HR managers input profile data through the frontend.
- The backend stores the information in the database and confirms the update to the user interface.

4.1.3 Entities

- **EmployeeProfile**
 - EmployeeID
 - Name
 - Department
 - Role
 - ContactDetails

4.2 Performance Review Module

4.2.1 Features

- Managers can schedule and evaluate employee performance.
- Review performance based on pre-defined criteria such as productivity, teamwork, and skill proficiency.

4.2.2 Entities

- **PerformanceReview**
 - ReviewID
 - EmployeeID
 - ManagerID
 - Date
 - PerformanceScore
 - Feedback

4.3 Feedback Collection Module

4.3.1 Features

- Collect feedback from various sources like peers, subordinates, and managers.
- Consolidate and analyze feedback data for a complete performance profile.

4.3.2 Entities

- **Feedback**
 - FeedbackID
 - FromEmployeeID
 - ToEmployeeID
 - FeedbackType (Peer, Manager, Subordinate)
 - Comments

4.4 Goal Management Module

4.4.1 Features

- Set and track personal and professional development goals.
- Track progress towards achieving goals, including milestone updates.

4.4.2 Entities

- **Goal**
 - GoalID

- EmployeeID
- GoalDescription
- TargetDate
- ProgressStatus

4.5 Reporting and Analytics Module

4.5.1 Features

- Generate reports on employee performance, feedback, and goal achievement.
- Provide insights and trends over time.

4.5.2 Entities

- **Report**
 - ReportID
 - EmployeeID
 - ReportDate
 - PerformanceSummary
 - FeedbackSummary

5. Deployment Strategy

5.1 Local Deployment

- The system can be initially deployed in a local environment for initial testing and review by the HR team.

5.2 Testing Environments

- Use containerized setups for staging environments, simulating production behavior.

6. Database Design

6.1 Tables and Relationships

- **EmployeeProfile**: Primary Key: EmployeeID.
- **PerformanceReview**: Foreign Key: EmployeeID, ManagerID.
- **Feedback**: Foreign Key: FromEmployeeID, ToEmployeeID.

- **Goal:** Foreign Key: EmployeeID.
- **Report:** Foreign Key: EmployeeID.

7. User Interface Design

7.1 Wireframes

- **Dashboard:** Displays employee profile, performance reviews, and progress on goals.
- **Performance Reviews:** Lists all performance reviews and feedback.
- **Goal Management:** Allows employees and managers to set, track, and update goals.

8. Non-Functional Requirements

8.1 Performance

- The system should be able to handle data updates for 500 employees at a time without performance degradation.

8.2 Usability

- User-friendly interface designed for HR professionals, managers, and employees with no technical background.

8.3 Security

- Access control mechanisms to ensure only authorized personnel (HR, managers, employees) can view or modify relevant data.

8.4 Scalability

- The system should be able to handle an increasing number of employees and performance reviews without requiring significant re-engineering.

9. Assumptions and Constraints

9.1 Assumptions

- Employees and managers will have internet access to use the platform.

9.2 Constraints

- The initial rollout will be limited to a specific department or location.