Employee Management System (EMS)

The **Employee Management System** is designed to streamline and automate essential HR and managerial tasks, including employee data management, salary adjustments, and notifications. With a structured approach, this system keeps all employee-related information organized and accessible for HR and managers, making it easy to handle payroll, onboarding, promotions, and offboarding. The system offers comprehensive employee profiles that capture crucial details, such as contact information, role and salary. This setup supports managers and HR teams in monitoring each employee's contributions, making informed decisions on salary adjustments, and facilitating promotions.

The Employee Management System is built following **SOLID principles**, which ensure each component operates independently yet integrates seamlessly with the rest of the system. This approach enhances modularity, making the code easy to extend and maintain over time.

Single Responsibility Principle (SRP) :

Each class in the system has a single, well-defined responsibility. For example: **Employee**, **FullTimeEmployee**, and **PartTimeEmployee** classes manage individual employee data.**HR** manages salary adjustments, payroll tasks, and employee onboarding. **Manager** focuses on promotions, performance evaluations, and salary approval.

By keeping these roles distinct, each part of the system remains focused on its task, making the code more manageable and easier to understand.

Open/Closed Principle (OCP) :

The system is designed to be open to extension but closed to modification:

New employee types (e.g., **ContractEmployee**) can be added by creating subclasses of Employee without altering the existing classes. New notification channels (e.g., **PushNotification**) can be implemented by creating new classes that implements the **Notifier** interface.

Liskov Substitution Principle (LSP) :

The system allows different employee types (e.g., **FullTimeEmployee**, **PartTimeEmployee**) to be used interchangeably:

Each subclass of **Employee** (e.g., **FullTimeEmployee, PartTimeEmployee**) can be managed by the same HR and Manager classes. This allows for flexible management of various employee types under a unified structure.

Interface Segregation Principle (ISP) :

The system defines specific interfaces for different functionalities:

The **Notifier** interface provides a single send method for sending notifications, allowing for separate implementations like EmailNotification and SMSNotification.

Dependency Inversion Principle (DIP):

The system's high-level modules depend on abstractions rather than concrete implementations:

EmployeeService depend on interfaces (**EmployeeDataAccess**) rather than concrete classes, allowing for flexibility in replacing or extending the data storage.

This design allows different components to be interchanged easily, enabling further expansion of the system without major changes to its structure.