

SOLID

1. Single Responsibility Principle (SRP)

- Each class in the design focuses on a single responsibility.
- Project focuses on handling project-related functionalities like managing tasks and team members.
- Task as well as its subclasses RecurringTask and PriorityTask focuses on handling task-specific details.
- TeamMember and its subclass ProjectManager handles details related to team members.

2. Open/Closed Principle (OCP)

- The design is open for extension but closed for modification.
- New Task types can be added by creating new subclasses such as RecurringTask and PriorityTask without needing to modify the existing code.

3. Liskov Substitution Principle (LSP)

- The object recurringSchedule in class RecurringTask can be implemented in class Task while maintain application logic

GRASP

4. Information Expert

- Classes are designed so that they have the information needed to do their responsibilities.
- Project knows about its tasks and team members.
- Task knows its title, description, due date, status, and priority.
- TeamMember knows its name, email, and role.

5. Creator

- The Project class is responsible for creating tasks as well as adding and removing team members.

6. High Cohesion

- Each class has a clear and focused responsibility.
- Project manages project-related functionalities.
- Task handles task-specific details.
- TeamMember manages team member information.