

Actividad 1: Análisis y Selección de Principios de Diseño

Para diseñar la arquitectura de esta plataforma, es necesario aplicar practicas de desarrollo que aseguran la modularidad y escalabilidad del código. Los principios que se aplicaran son los siguientes:

1. Principios S.O.L.I.D:

- a. Single Responsibility Principle (SRP): Cada componente de la arquitectura debe tener una responsabilidad única. Facilita la modularidad y el mantenimiento.
- b. Open/Closed Principle (OCP): Los módulos deben de estar abiertos para su extensión, es decir, que su comportamiento puede extenderse para añadir nuevas funcionalidades, pero cerrados a la modificación. Esto se refiere a que si el modulo funciona, no se debería alterar el código existente al añadir nuevas utilidades.
- c. Liskov Sustitution Principle (LSP): Las clases “hijas” de una clase deberán poder sustituir a sus clases “padre” sin alterar el funcionamiento del programa. Esto es porque las clases derivadas siempre deben ser una extensión de las clases base, no una alteración de estas.
- d. Interface Segregation Principle (ISP): Se evita usar una interfaz monolítica y general, favoreciendo interfaces con funcionalidades específicas.
- e. Dependency Inversión Principle (DIP): Las entidades de software deben de depender de abstracciones, no de implementaciones. Es decir, en vez de referenciar módulos entre sí, usar referencias a interfaces.

2. Don't Repeat Youreself (DRY): El código se reutilizará mediante la abstracción y modularizacion, no mediante la duplicación del mismo código en múltiples componentes.

3. Keep it Simple, Stupid (KISS): Se debe mantener el código simple y comprensible, evitando la sobreingeniería.
4. You ain't Gonna Need it (YAGNI): Se implementarán solo las utilidades necesarias, y se evitará agregar características que creamos que nos puedan ser útiles en un futuro hasta que estas sean verdaderamente necesarias.

La aplicación de estas estrategias nos garantizará que esta plataforma de gestión de proyectos sea mantenible, flexible, y escalable. Podremos crear una arquitectura sólida que nos facilite adaptarla a futuras evoluciones de la plataforma.