



UTT

UNIVERSIDAD TECNOLÓGICA DE TIJUANA

GOBIERNO DE BAJA CALIFORNIA

Actividad:

Selection of Design Patterns

Estudiante:

Ángel Amaury Tienda Lezama

Grupo:

10B

Materia:

Desarrollo móvil Integral

Docente:

Ray Brunett Parra Galaviz

Tijuana, Baja California, 07 de Enero del 2025

Introducción

Los patrones de diseño son soluciones reutilizables a problemas comunes en el desarrollo de software. Actúan como guías que permiten a los desarrolladores crear sistemas más flexibles, escalables y fáciles de mantener. La selección del patrón de diseño adecuado depende del problema que se desea resolver y de las características del proyecto. Utilizar patrones correctamente puede optimizar el tiempo de desarrollo y garantizar un código más estructurado y comprensible.

I. Comprender el Problema

Antes de seleccionar un patrón, es esencial identificar el problema o la necesidad específica que se quiere abordar.

Ejemplos de problemas comunes:

- ¿Cómo instanciar objetos sin acoplar el código? (Problemas creacionales).
- ¿Cómo estructurar y organizar relaciones entre clases y objetos? (Problemas estructurales).
- ¿Cómo definir la comunicación y el comportamiento entre objetos? (Problemas comportamentales).

II. Categorías Principales de Patrones de Diseño

Los patrones se agrupan en tres categorías, según el tipo de problema que resuelven:

Patrones Creacionales:

- Se centran en cómo crear objetos de manera flexible.
- Cuándo usarlos: Cuando la creación de objetos es compleja o debe desacoplarse de su implementación.

Patrones Estructurales:

- Abordan cómo organizar y estructurar relaciones entre clases y objetos.
- Cuándo usarlos: Cuando se necesita flexibilizar las relaciones entre clases y componentes.

Patrones Comportamentales:

- Se enfocan en cómo los objetos interactúan y comunican entre sí.
- Cuándo usarlos: Cuando se necesita gestionar la comunicación entre objetos o encapsular comportamientos.

III. Análisis de Requisitos del Proyecto

Evaluar los requisitos funcionales y no funcionales del sistema para determinar qué patrones son necesarios.

Factores clave a considerar:

- Complejidad del sistema.
- Escalabilidad requerida.
- Mantenibilidad del código.
- Flexibilidad para cambios futuros.

IV. Consideraciones para Implementar Patrones

Simplicidad: No aplicar patrones si no son necesarios; evita el exceso de complejidad.

Reutilización: Un patrón bien implementado debe ser reutilizable en diferentes contextos.

Compatibilidad: Asegurarse de que el patrón seleccionado se alinee con las tecnologías y frameworks utilizados.

V. Evaluación y Ajustes

Una vez implementado el patrón, evaluar si cumple con los objetivos establecidos.

Recopilar retroalimentación de los desarrolladores y analizar su impacto en la mantenibilidad y escalabilidad del sistema.

Conclusión

La selección de patrones de diseño es un proceso estratégico que depende del análisis detallado del problema y los requisitos del sistema. Elegir el patrón correcto no solo facilita la resolución del problema, sino que también mejora la calidad y la flexibilidad del código. Sin embargo, es importante aplicar patrones de manera pragmática, evitando complicar innecesariamente el diseño del sistema. Un enfoque equilibrado garantizará un desarrollo eficiente y un software robusto.