

Segundo Principio SOLID

Principio de OCP (Open/Closed Principle)

Segundo dentro de los principios SOLID, establece que un componente de software debe estar abierto para la extensión, pero cerrado para la modificación. Esto significa que, una vez que una clase, módulo o servicio ha sido probado y validado, no debería ser necesario modificarlo cada vez que surjan nuevos requerimientos. En lugar de eso, el diseño debe permitir agregar nuevas funcionalidades mediante la creación de nuevas clases o implementaciones, evitando así tocar el código que ya funciona correctamente.

Este principio busca proteger la estabilidad del sistema. Cada vez que un desarrollador modifica una pieza de código ya existente, corre el riesgo de introducir errores o efectos colaterales en otras partes del sistema. Al seguir el OCP, se reduce esa posibilidad porque el código base permanece inmutable y se aprovechan mecanismos como la herencia, las interfaces y el polimorfismo para añadir comportamientos adicionales.

Ciclo del Principio Abierto-Cerrado

