

**REFACTORING  
· GURU ·****★ Contenido premium****Patrones de diseño**

Introducción

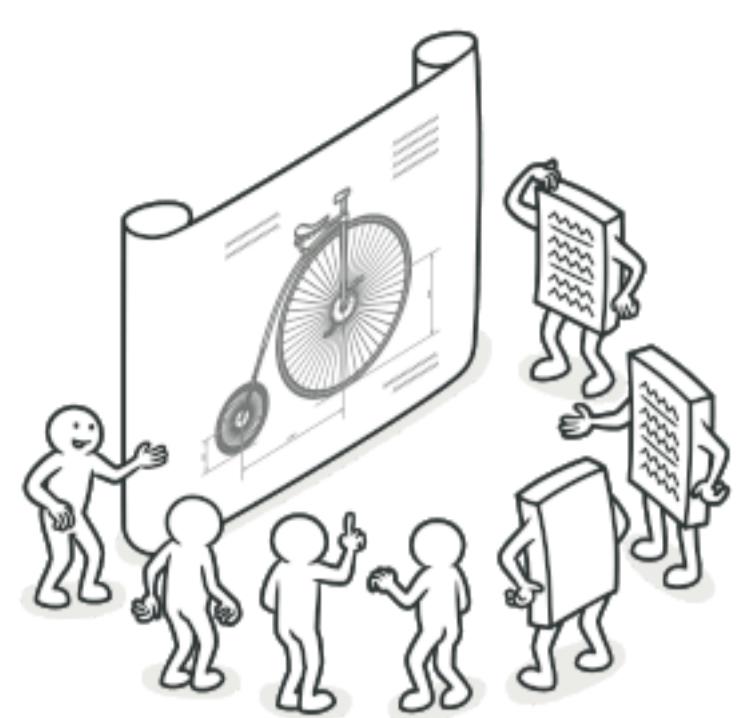
Catálogo

Patrones creacionales

Patrones estructurales

Patrones de comportamiento

Ejemplos de código

**Refactorización**  
(próximamente)[Iniciar sesión](#)[Contáctanos](#)

# PATRONES de DISEÑO

Los **patrones de diseño** (design patterns) son soluciones habituales a problemas comunes en el diseño de software. Cada patrón es como un plano que se puede personalizar para resolver un problema de diseño particular de tu código.

[¿Qué es un patrón de diseño?](#)

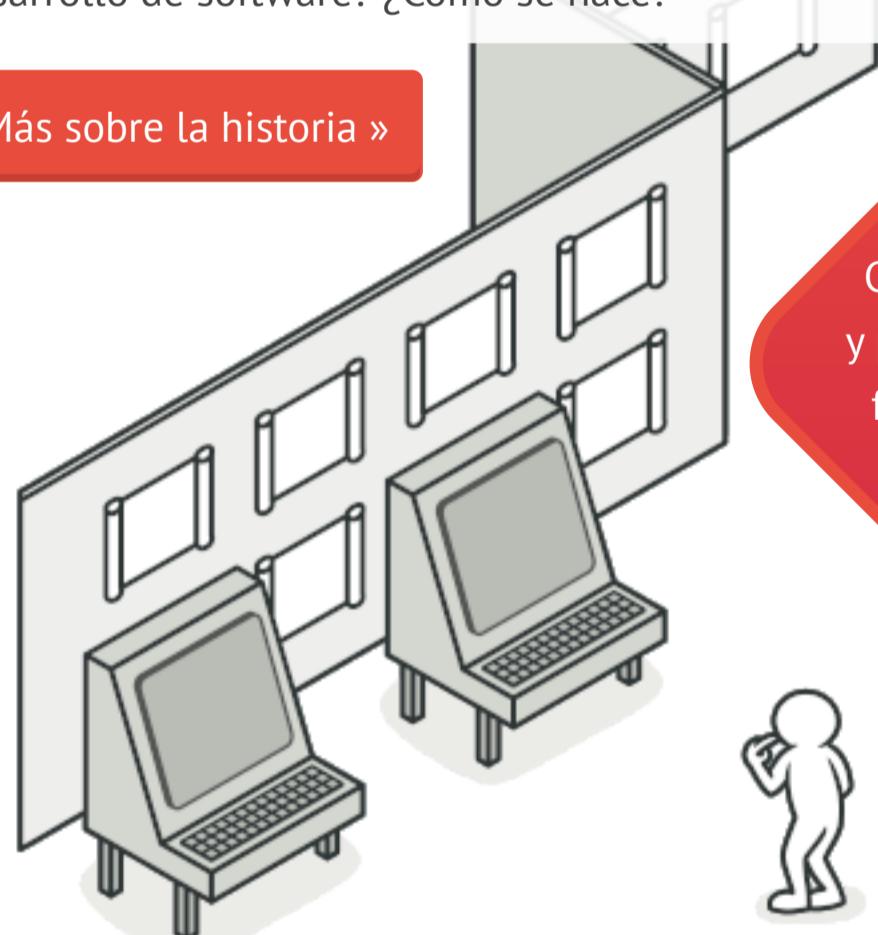
## Ventajas de los patrones

Los patrones son un juego de herramientas que brindan soluciones a problemas habituales en el diseño de software. Definen un lenguaje común que ayuda a tu equipo a comunicarse con más eficiencia.

[Más sobre las ventajas »](#)

## Historia de los patrones

¿Quién inventó los patrones y cuándo?  
¿Se pueden utilizar los patrones fuera del desarrollo de software? ¿Cómo se hace?

[Más sobre la historia »](#)

## Sumérgete en los **PATRONES DE DISEÑO**

Consulta nuestro ebook sobre patrones y principios de diseño. Está disponible en formatos PDF/ePUB/MOBI e incluye el archivo con ejemplos de código en Java, C#, C++, PHP, Python, Go, Ruby, TypeScript y Swift.

[★ Saber más sobre el libro](#)

## Crítica de los patrones

¿Son tan buenos los patrones como se dice?  
¿Es siempre posible utilizarlos?  
¿Pueden los patrones ser dañinos en alguna ocasión?

[Más sobre la crítica »](#)