



**DigitalHouse** >  
Coding School

# Patrón facade



**Certified Tech  
Developer**  
The Ultimate Degree

“

Vamos a conocer cómo hacemos para reducir la complejidad de interactuar con un conjunto de subsistemas, proporcionando un único punto de entrada.



”

## Propósito



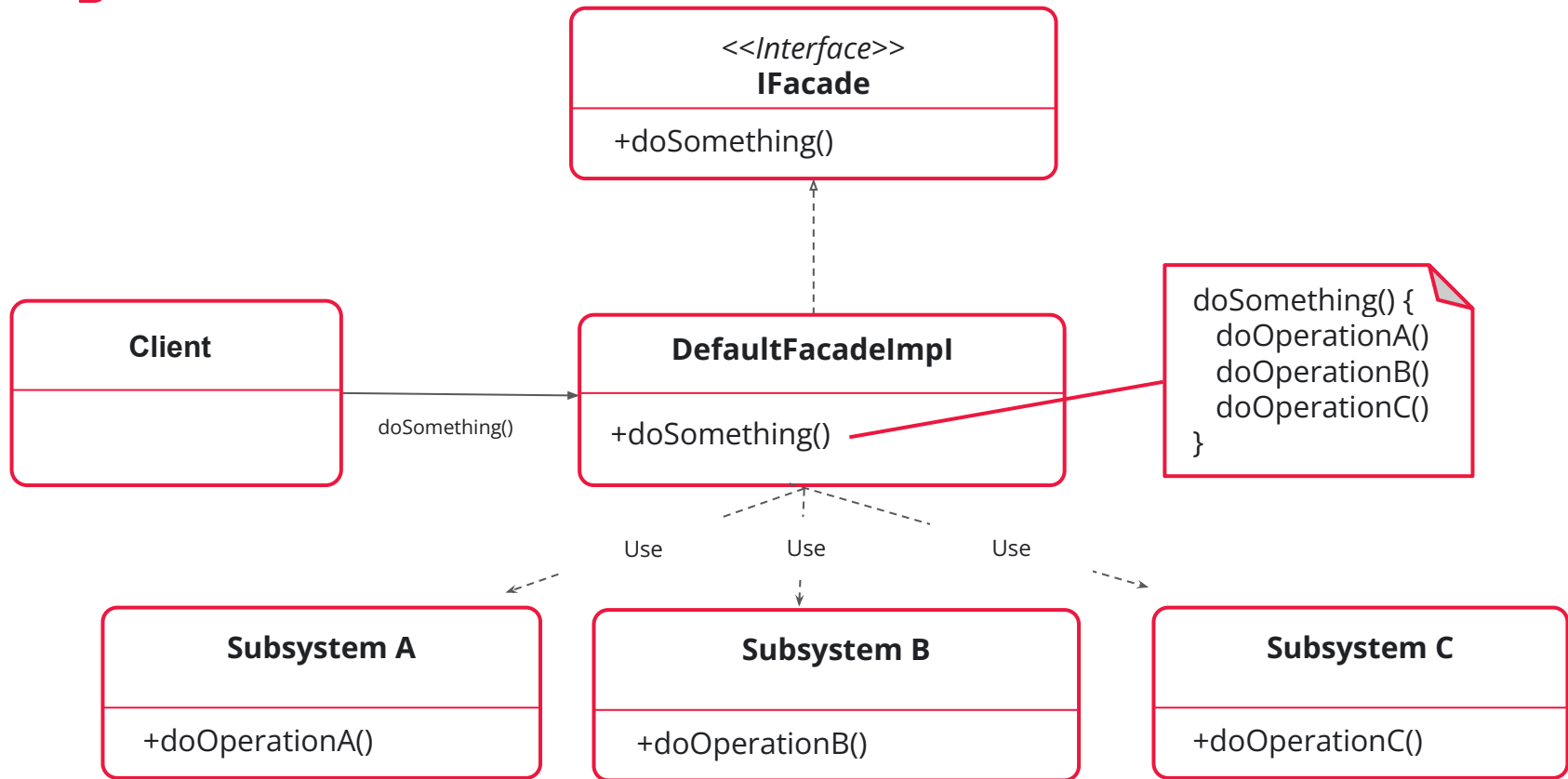
Facade es un patrón de diseño estructural que proporciona una interfaz simplificada a una biblioteca, un framework o cualquier otro grupo complejo de clases.

## Solución



Se dispone de una interfaz que define como el cliente se va a comunicar con el sistema. Esta clase implementará esta interfaz para poder recibir las peticiones y será la encargada de enviarle la petición del cliente a la clase que corresponda (subsistemas).

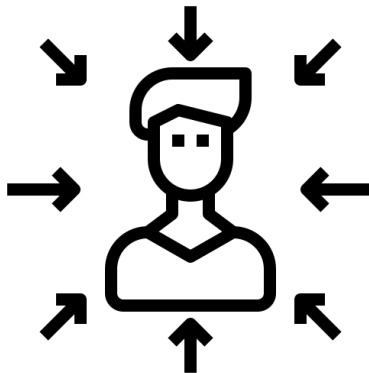
# Diagrama UML



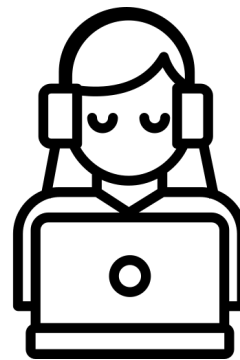
# Ventajas



El *software* se vuelve más flexible y fácilmente de expandir.

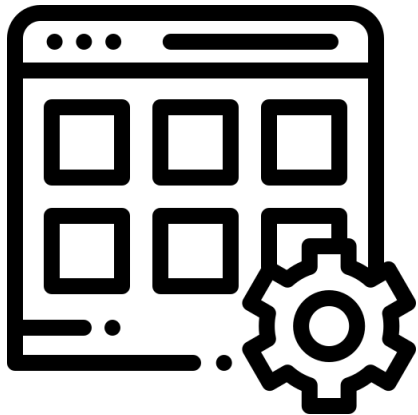


Reducimos el uso de objetos que tratan directamente con el cliente, haciendo que el sistema sea más fácil de usar.



Reducimos el acoplamiento entre el cliente y los subsistemas. Esto nos permite modificar los subsistemas sin afectar al cliente.

# Desventajas



Alto grado de dependencia en la interfaz de la fachada.

**DigitalHouse** >  
Coding School



**Certified Tech  
Developer**

The Ultimate Degree