Patrón cadena de responsabilidad

DigitalHouse >
Coding School

Certified Tech Developer
The Ultimate Degree

Veamos cómo podemos resolver problemas cuando no estamos muy seguros de qué objeto deberá procesar una solicitud concreta.

Propósito y solución

Propósito

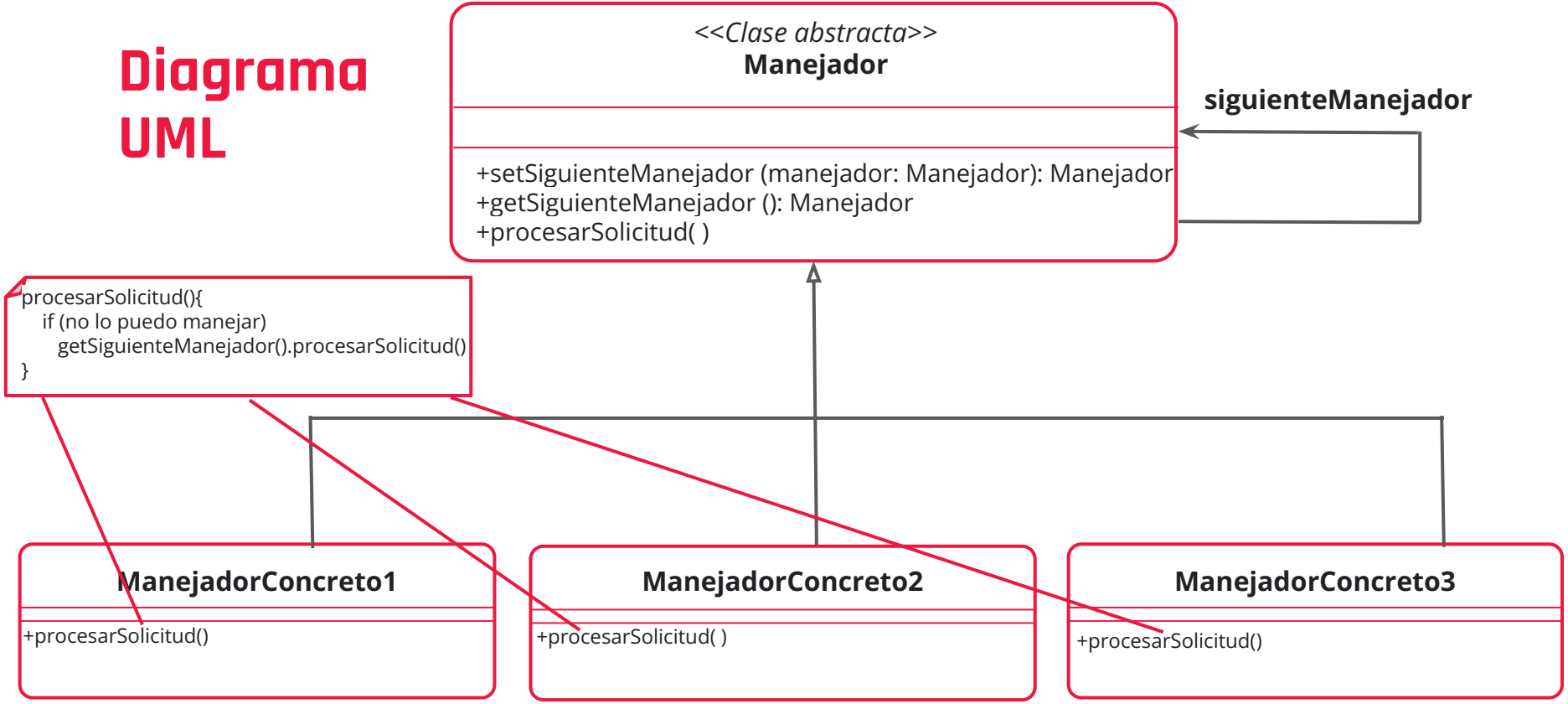
Es un patrón de diseño de comportamiento que permite pasar solicitudes a lo largo de una cadena de manejadores.

Al recibir una solicitud, cada uno de ellos, decide si la procesa o la pasa al siguiente manejador.

Solución

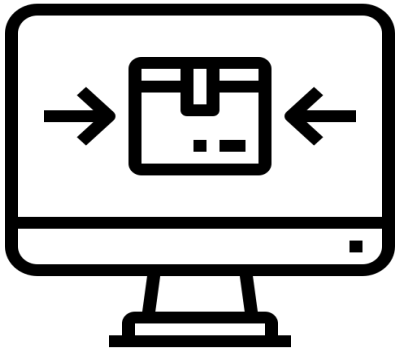
Crear una cadena con las **clases manejadores** para que procesen la solicitud del cliente. Cada uno tiene un campo para almacenar una referencia al siguiente manejador de la cadena.
La solicitud viaja por la misma hasta que todos los manejadores hayan tenido la oportunidad de procesarla
(los manejadores pueden decidir no pasar la solicitud y detener el procedimiento).

Diagrama UML



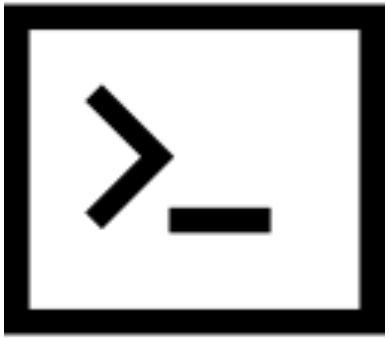
Ventajas

Mayor flexibilidad para procesar las peticiones del cliente. Es posible agregar objetos que sepan resolver nuevas responsabilidades o modificar las actuales sin afectar al cliente.



Menor acoplamiento
Permite que un objeto envíe una petición y sepa que va a ser tratada, pero tanto el emisor como el receptor no conocen nada del otro.

Desventajas



Puede ser complejo implementar la cadena, y si no está bien configurada, puede que no se cubran todas las peticiones.


DigitalHouse>
Coding School


Certified Tech
Developer
The Ultimate Degree