



Chain of Responsibility pattern

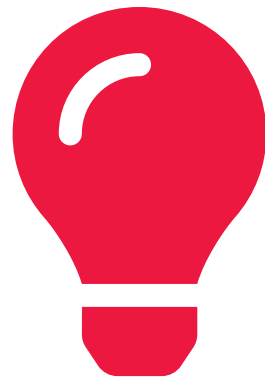


**Certified
Developer**
The Ultimate Tech Degree

DigitalHouse >
Coding School



Vamos ver como podemos resolver problemas quando não temos certeza de qual objeto deve processar uma solicitação específica.





Propósito e solução

Propósito



É um design pattern comportamental que permite que as solicitações passem por uma corrente de manipuladores.

Ao receber uma solicitação, cada um deles decide se vai processá-la ou repassá-la ao próximo manipulador.

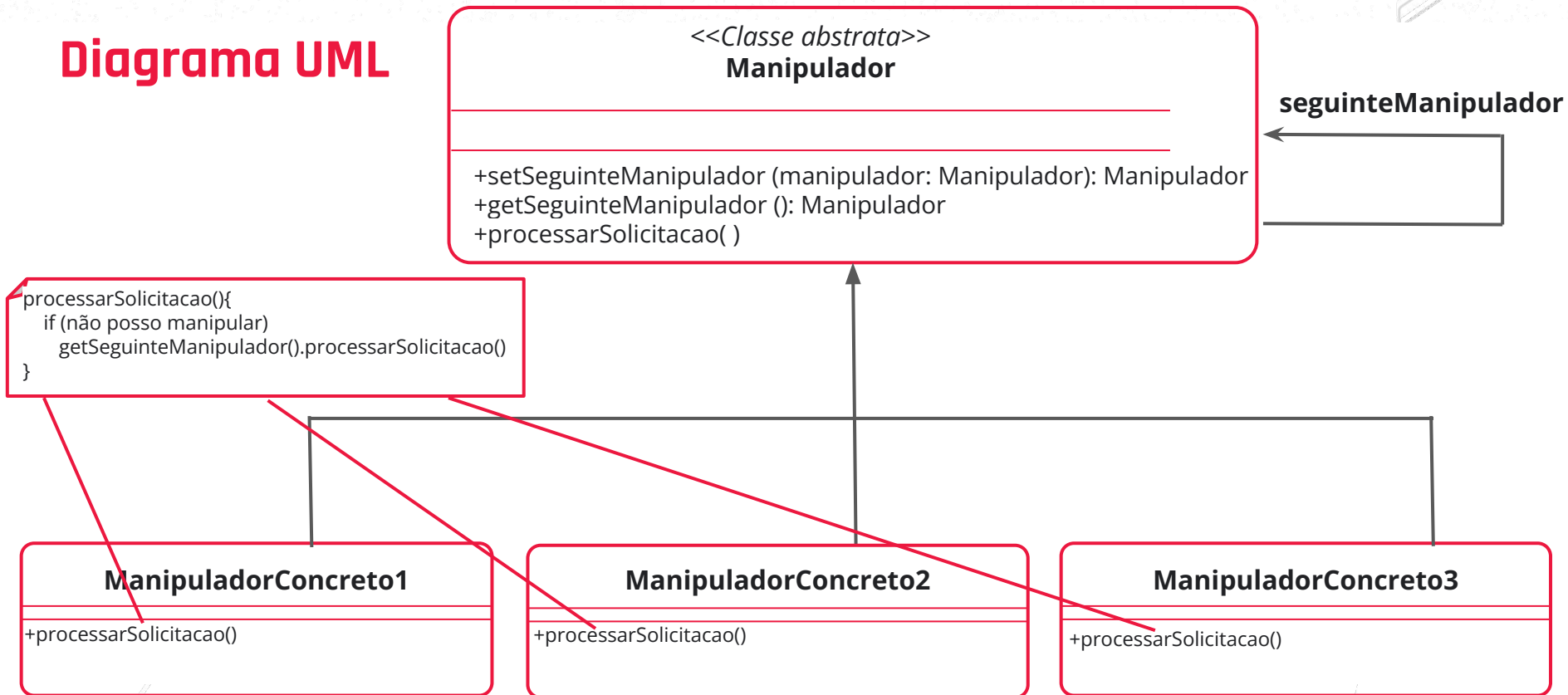
Solução



Criar uma corrente com as classes manipuladoras para processar a solicitação do cliente. Cada uma possui um campo para armazenar uma referência ao próximo manipulador na corrente. A solicitação viaja até que todos os manipuladores tenham a chance de processá-la (Os manipuladores podem decidir não aprovar a solicitação e interromper o procedimento).

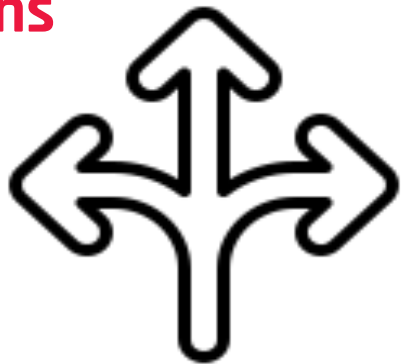


Diagrama UML

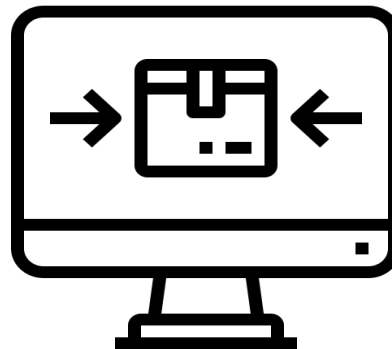




Vantagens



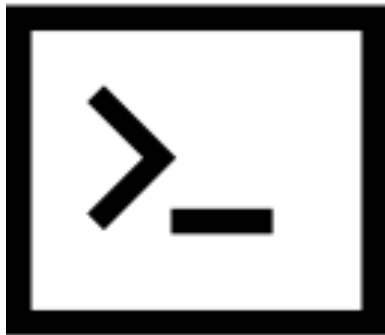
Maior flexibilidade para processar as solicitações dos clientes. É possível agregar objetos que saibam resolver novas responsabilidades ou modificar as atuais sem afetar o cliente.



Menor acoplamento:
Permite que um objeto envie uma solicitação e saiba que ela será processada, mas tanto o remetente quanto o destinatário não sabem nada sobre o outro.



Desvantagens



Pode ser complexo implementar a corrente e, se não for configurada corretamente, todas as solicitações podem não ser atendidas.



DigitalHouse >
Coding School



**Certified Tech
Developer**

The Ultimate Degree