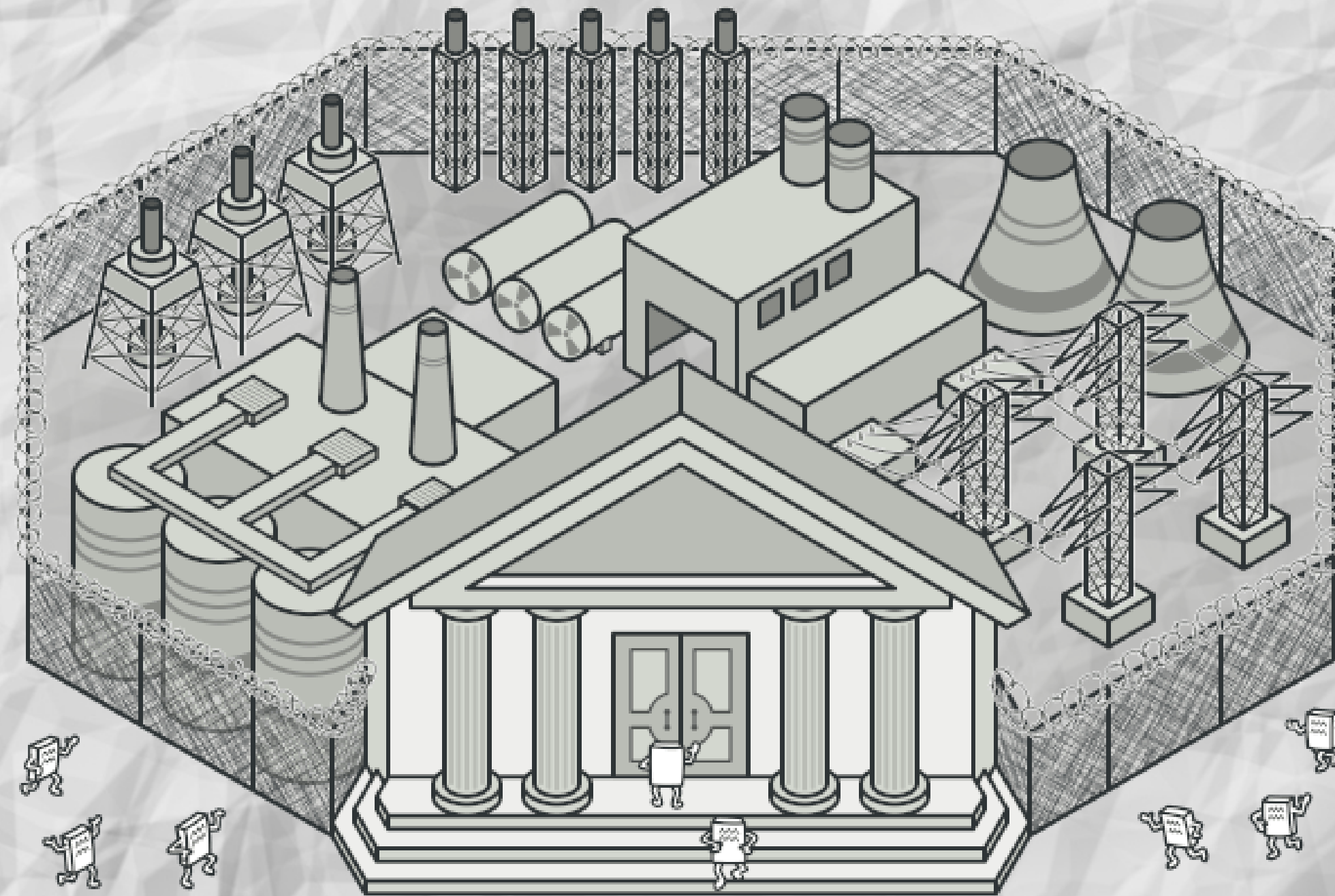


Facade Pattern



Integrantes: Pierre Iost, Alex Macedo, Anderson Gonçalves

Introdução ao padrão de projeto: Facade

Problema

Imagine que você precisa fazer seu código funcionar com um amplo conjunto de objetos que pertencem a uma sofisticada biblioteca ou framework. Normalmente, você precisaria inicializar todos aqueles objetos, rastrear as dependências, executar métodos na ordem correta, e assim por diante.



Uma facade é uma classe que fornece uma interface simples para um subsistema complexo que contém muitas partes que se movem.

Ter uma facade é útil quando você precisa integrar sua aplicação com uma biblioteca sofisticada que tem dúzias de funcionalidades, mas você precisa de apenas um pouquinho delas.



Prós e contras de uma Facade

Prós

Uma fachada pode fornecer funcionalidades limitadas em comparação com trabalhar com os subsistemas diretamente. Contudo, ela inclui apenas aquelas funcionalidades que o cliente se importa.

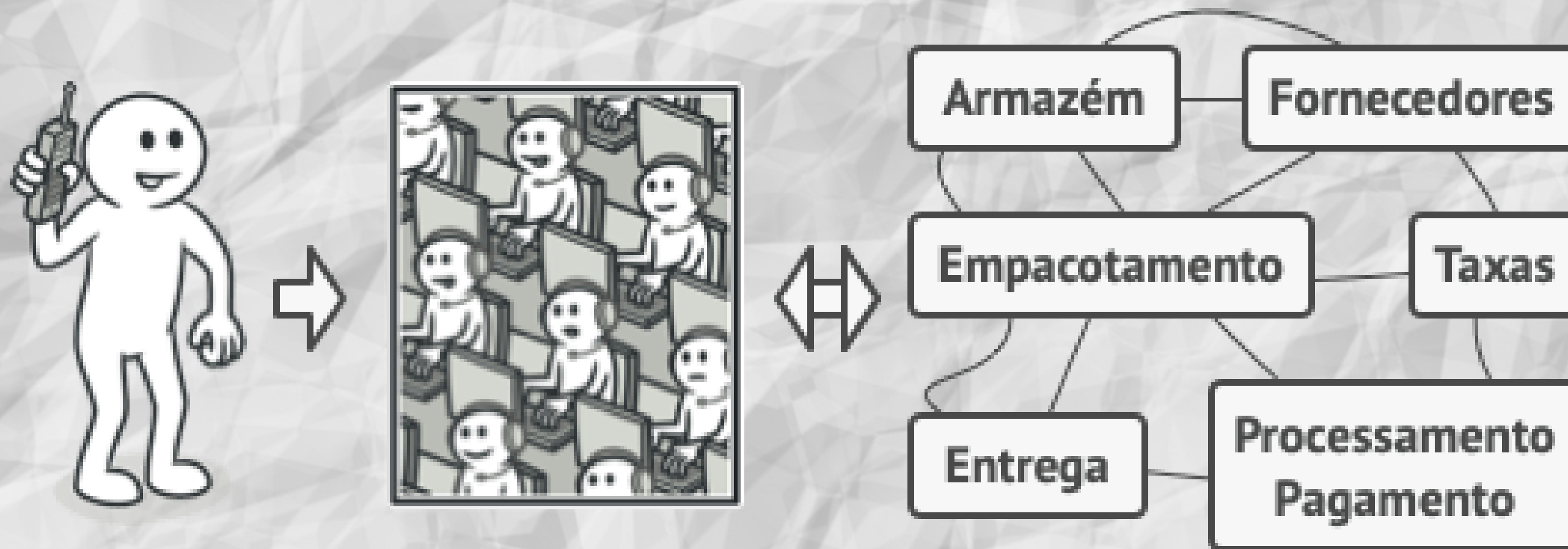
O Padrão Facade nos permite desconectar a implementação do cliente de qualquer subsistema. Assim, se quiséssemos acrescentar novas funcionalidades no subsistema seria necessário apenas alterar a Facade ao invés de alterar diversos pontos do sistema. Além disso, o padrão Facade simplifica uma interface tornando-a muito mais simples e unifica um conjunto de classes mais complexas que pertencem a um subsistema.

Contras

Uma fachada pode se tornar um objeto deus acoplado a todas as classes de uma aplicação, sendo presente em todas as instâncias do código.

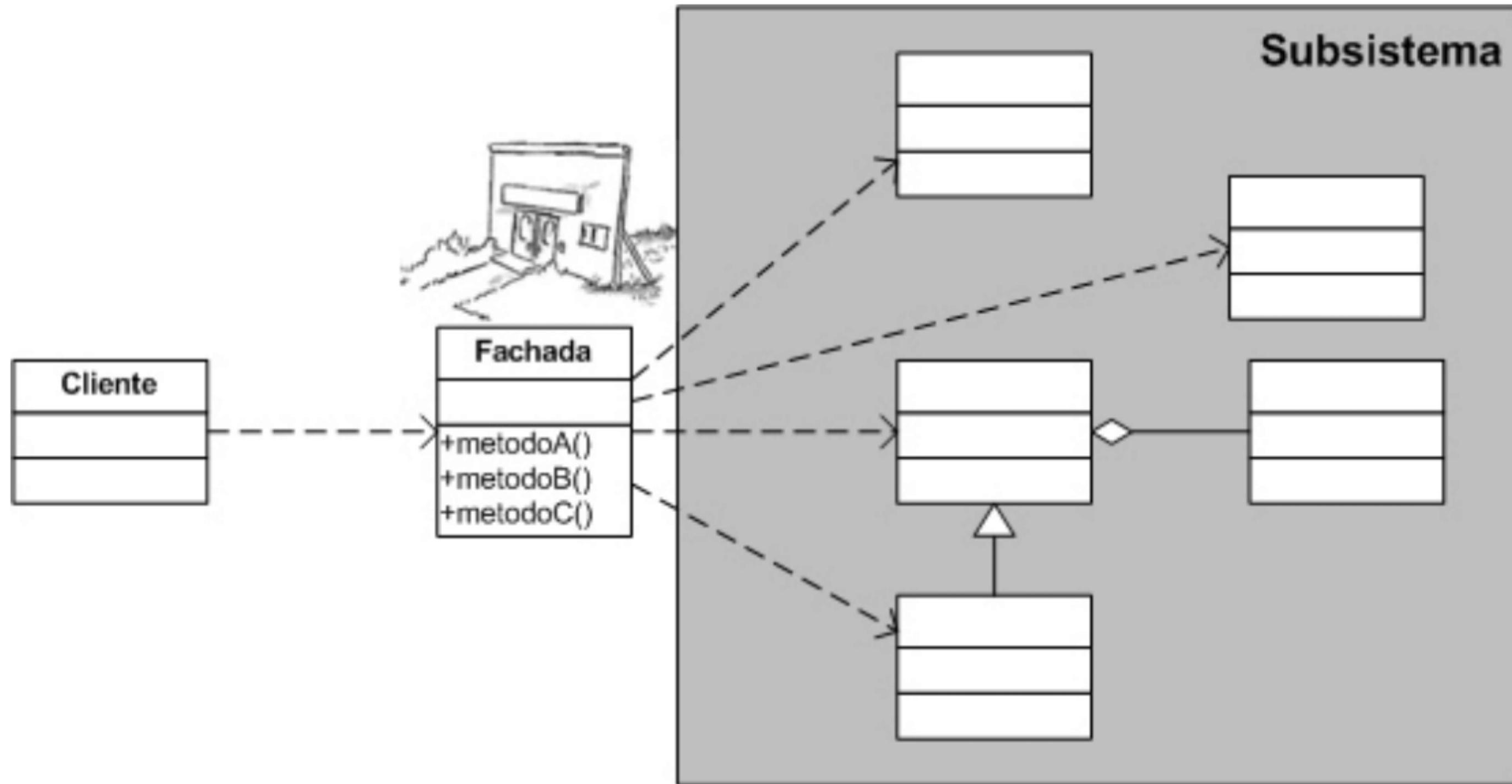
Mexer na facade para resolver um problema pode acarretar problemas em outros lugares do código.

Analogia com mundo real

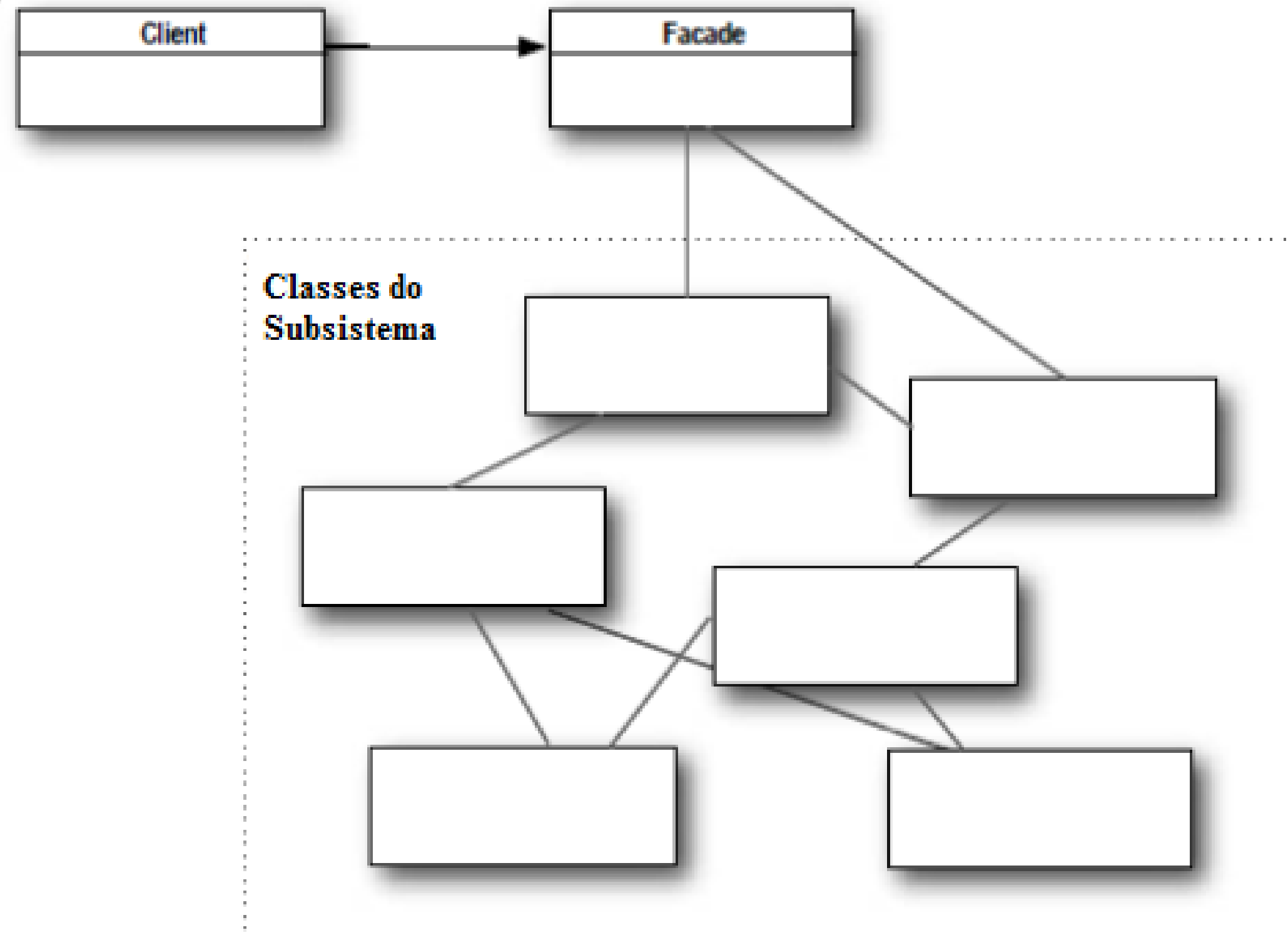


Quando você liga para uma loja para fazer um pedido, um operador é sua fachada para todos os serviços e departamentos da loja. O operador fornece a você uma simples interface de voz para o sistema de pedido, pagamentos, e vários sistemas de entrega.

Como nota-se, o usuário ira ver somente a facade, e dentro dela terá o resto do sistema



Outro exemplo de diagrama que representa bem uma facade



Conclusões

O Padrão Facade é utilizado quando precisamos simplificar e unificar uma interface grande ou um conjunto complexo de interfaces. Uma das vantagens do padrão Facade é desconectar o cliente de um subsistema complexo, conforme pode ser visto no diagrama de classes. Um sistema pode ter diversos Facades simplificando diversos pontos do programa.

Integrantes: Pierre Iost, Alex Macedo, Anderson Gonçalves