

Padrões de Projetos

Introdução

- Nas últimas décadas, ocorreu na indústria de engenharia de software um enorme progresso no campo dos padrões de projeto
- Arquiteturas testadas para construir softwares orientados a objetos flexíveis e sustentáveis.
- Utilizar padrões de projeto reduz substancialmente a complexidade do processo de design.
- Projetar um sistema de controle de tráfego aéreo será uma tarefa menos complexa se

Introdução

- Como os padrões de projeto beneficiam os desenvolvedores de um sistema:
 - **ajudando a construir um software confiável com arquiteturas testada e perícia acumulada pela indústria.**
 - **promovendo a reutilização de projetos em futuros sistemas.**
 - **Ajudando a identificar equívocos comuns e**

Introdução

- O conceito de utilização de padrões de projeto para construir sistemas de softwares originados no campo da arquitetura.
- Arquitetos utilizam uma série de elementos de projetos arquitetônicos estabelecidos, como arcos e colunas, ao projetarem edifícios.
- Projetar com arcos e colunas é uma estratégia testada para construir edifícios perfeitos — esses elementos podem ser

Introdução

- Nos softwares, os padrões de projeto não são classes nem objetos.
- Em vez disso, os projetistas utilizam padrões de projeto para construir conjuntos de classes e objetos.
- Para utilizar padrões de projeto eficientemente, os projetistas devem conhecer os padrões mais famosos e eficientes utilizados na indústria de engenharia de software.

O histórico dos padrões de projeto orientados a objetos

- Entre 1991 e 1994, Erich Gamma, Richard Helm, Ralph Johnson e John Vlissides — coletivamente conhecidos como ‘Gang of Four’ — utilizaram suas perícias para escrever o livro “Design patterns: elements of reusable object-oriented software”
- Esse livro descreve 23 padrões de projeto, cada um fornecendo uma solução a um problema comum de projeto de software na indústria.
- O livro agrupa os padrões de projeto em três categorias:
 - padrões de projeto criacionais
 - padrões de projeto estruturais
 - padrões de projeto comportamentais

Padrões de Projeto Criacionais

- Padrões de projeto criacionais descrevem as técnicas para instanciar objetos (ou grupos de objetos).
- Padrões de projeto estruturais permitem que os projetistas organizem classes e objetos em estruturas maiores.
- Padrões de projeto comportamentais atribuem responsabilidades a classes e objetos.

Padrões de Projeto Criacionais

- GoF define 5 padrões criacionais:
 - Abstract Factory
 - Builder
 - Factory Method (Seção M.3)
 - Prototype (Seção M.6)
 - Singleton (Seção M.2)

Padrões de projeto Criacionais

- Examinam questões relacionadas à criação de objetos.
- por exemplo, impedir que um sistema crie mais de um objeto de uma classe (o padrão de projeto criacional Singleton)
- Ou postergar, até o tempo de execução, a decisão sobre quais tipos de objetos serão criados

Criacionais: Singleton

- Ocasionalmente, um sistema deve conter um único objeto de uma classe depois que o programa instancia esse objeto, ele não deve ter permissão de criar objetos adicionais dessa classe.
- Por exemplo, alguns sistemas são conectados a um banco de dados utilizando apenas um objeto que gerencia as conexões ao banco de dados, isso assegura que outros objetos não inicializem conexões desnecessárias que tornariam o sistema lento.
- Padrão de projeto Singleton garante que um sistema instancie no máximo um objeto de uma classe.

Padrões de projeto Estruturais

- Descrevem maneiras comuns de organizar classes e objetos em um sistema. O livro da Gang of Four descreve sete padrões de projeto estruturais:
- Adapter
- Bridge
- Composite
- Decorator
- Facade
- Flyweight
- Proxy

Adapter

- O padrão de projeto Adapter fornece a um objeto uma nova interface que se adapta à interface de um outro objeto, permitindo que os dois objetos colaborem entre si.
- Poderíamos equiparar o padrão adapter a um adaptador de tomada em um dispositivo elétrico — soquetes elétricos na Europa têm uma forma diferente daqueles nos Estados Unidos, portanto, é necessário um adaptador para conectar um dispositivo norte-americano a um soquete europeu e vice-versa

Adapter

- O Java fornece várias classes que utilizam o padrão de projeto Adapter.
- Os objetos das subclasses concretas dessas classes atuam como adaptadores entre objetos que geram certos eventos e objetos que tratam os eventos.
- Por exemplo, um MouseAdapter, que explicamos na Seção 11.14, adapta um objeto que gera MouseEvents para um objeto que trata MouseEvents

Padrões de projeto comportamentais

- Fornecem estratégias testadas para modelar a maneira como os objetos colaboram entre si em um sistema e
- Oferecem comportamentos especiais apropriados para uma ampla variedade de aplicativos

Padrões de projeto comportamentais

- O livro da Gang of Four descreve 11 padrões de projeto comportamentais :

Padrões de projeto comportamentais

- Chain of Responsibility
- Command
- Interpreter
- Iterator
- Mediator
- Memento
- Observer
- State
- Strategy
- Template Method
- Visitor

Chain of Responsibility

- Em sistemas orientados a objetos, os objetos interagem por meio do envio de mensagens. Frequentemente, um sistema precisa determinar em tempo de execução o objeto que tratará uma mensagem particular.
- Por exemplo, considere o projeto de um sistema de telefonia com três linhas para um escritório. Quando alguém chama o escritório, a primeira linha trata a chamada — se a primeira linha estiver ocupada, a segunda tratará a chamada e se a segunda estiver ocupada, a terceira tratará a chamada.
- Se todas as linhas no sistema estiverem ocupadas, uma secretária eletrônica irá instruir o chamador a esperar a próxima linha disponível. Quando uma linha estiver disponível, essa tratará a chamada

Chain of Responsibility

- permite que um sistema determine em tempo de execução o objeto que tratará uma mensagem. Esse padrão permite que um objeto envie uma mensagem para vários objetos em uma cadeia.
- Cada objeto na cadeia pode tratar a mensagem ou passá-la para o próximo objeto. Por exemplo, a primeira linha no sistema de telefonia é o primeiro objeto na cadeia de responsabilidades, a segunda linha é o segundo objeto, a terceira linha é o terceiro objeto e a secretária eletrônica é o quarto objeto.

Chain of Responsibility

- O objeto final na cadeia é a próxima linha disponível que trata a mensagem. A cadeia é criada dinamicamente em resposta à presença ou ausência de operadores específicos de mensagens.

Exercício

- Analise o texto no moodle e identifique que tipos de Padrões de Projeto podem ser aplicados a seu Trabalho Final
- Ao final iremos discutir em aula a escolha de cada grupo