

## - Atividade Avaliativa Padrões de Software

Nome: Vinícius Barbosa da Silva

- 1) Os Frameworks são tidos como facilitadores e permite com que Devs criem soluções/ implementem de maneira mais fácil e rápida, como os Frameworks são muito usados e quase necessários hoje em dia, fazê-los de qualquer forma não é interessante e futuramente pode deixar-lo extremamente difícil de ser utilizado. Visto que os Padrões são soluções já consolidadas e testadas em projetos anteriores, que podem ser aplicadas em novos projetos para resolver problemas recorrentes e melhorar a qualidade e a eficiência do código. Padrões de projeto podem ser muito úteis na construção de frameworks, pois permitem que os desenvolvedores projetem componentes e abstrações reutilizáveis, que atendam às necessidades de diferentes projetos. Os padrões de projeto também ajudam a estabelecer convenções e boas práticas de design, facilitando o uso do framework pelos desenvolvedores.
- 2) (implementação)
- 3) Template method atinge o objetivo usando herança, enquanto Strategy podemos dizer que é por delegação/composição. Se houver código em comum entre as implementação, Template method é uma boa alternativa, se não houver nada em comum, prefira Strategy.
- 4) (implementação)
- 5) (implementação)
- 6) (implementação)
- 7) (implementação)
- 8) Um método abstrato convencional é aquele que deve ser implementado pelas subclasses que estendem a classe abstrata que define o template. Já o método gancho (hook) é um método abstrato opcional que é definido na classe abstrata, mas não precisa ser implementado pelas subclasses. Ele é usado para permitir que as subclasses personalizem ou estendam o comportamento do algoritmo em etapas específicas, sem precisar reimplementar todo o template. O método gancho fornece um ponto de extensão para que a subclasse possa adicionar ou modificar o comportamento sem afetar o funcionamento geral do algoritmo.  
Um exemplo de uso de um método gancho pode ser para permitir que a subclasse defina um comportamento específico para uma etapa do algoritmo que pode ser opcional ou ter variações. Outro exemplo pode ser para permitir que a subclasse especifique um comportamento diferente dependendo do contexto ou estado em que o algoritmo está sendo executado. Os métodos gancho também podem ser usados para permitir que a subclasse modifique a ordem ou a frequência de execução de uma etapa do algoritmo.

- 9) Por exemplo, suponha que temos uma classe abstrata `Processador` que define um método `processar()` que segue uma sequência fixa de etapas para processar uma solicitação. Uma dessas etapas é "validação", que pode ser implementada de diferentes maneiras por diferentes subclasses de processador. Podemos usar o padrão Chain of Responsibility para criar uma cadeia de validação que lida com a validação de solicitações de forma sequencial. Cada objeto na cadeia de validação tem a oportunidade de validar a solicitação e passá-la para o próximo objeto na cadeia se a validação for bem-sucedida. Se a validação falhar, a cadeia pode interromper o processamento e retornar um erro. Assim, a cadeia de validação pode ser considerada um gancho (hook) dentro do método `processar()` do `Processador`, permitindo que diferentes implementações de validação sejam executadas em uma ordem específica, dependendo da solicitação e das necessidades do cliente.