

TPPE - Técnicas de Programação para Plataformas Emergentes - TP3

(07/02/2023)

Alunos:

Marcos Gabriel Tavares - 170041042

Luis Gustavo Ferreira Marques - 180105604

Características:

Simplicidade:

Essa característica é a mais importante para um código bem projetado. Um projeto simples é fácil de entender e não possui elementos desnecessários e é coerente, consistente e fácil de implementar, sendo o menor o possível. Sempre é possível adicionar mais código para funcionalidades extras, mas é difícil remover algo que já está entrelaçado. Para alcançar simplicidade deve-se projetar de forma que seu projeto relegue o máximo de trabalho possível, concentrando-se apenas nos problemas imediatos, criando programas que fazem muito com pouco.

Mau Cheiro :

Pode-se associar essa característica com a varios dos maus-cheiros devido a sua pervasividade, alguns deles são: **Classe inchada, método longo, código duplicado, aglomerados de dados, campo temporário, instruções switch, comentários, lista de parâmetros longa demais etc.** Como uma boa possível opção de refatoração tem-se Simplificar operações condicionais.

Modularidade:

Quando desenvolvemos um programa acabamos dividindo ele em partes chamadas módulos ou componentes. Decompomos esses módulos em subsistemas, bibliotecas, pacotes, classes e assim por diante. Cada parte é menos complexa que o problema original, mas juntas elas completam a solução. A qualidade dessa decomposição é primordial. As qualidade chave da modularidade são chamadas de coesão e acoplamento. Como desenvolvedores devemos almejar Alta Coesão e Baixo Acoplamento. Alta coesão significa que os módulos e as partes que compõem esses módulos funcionam bem juntas. Baixo acoplamento significa que os módulos devem depender o mínimo possível uns dos outros. Dessa forma os módulos podem ser desenvolvidos e testados por times diferentes.

Mau Cheiro :

Pode-se associar essa característica com os mau cheiros: **Mudanças divergentes, cirurgia com rifle, hierarquia de heranças paralelas, classe preguiçosa, classes alternativas com diferentes interfaces, cadeias de mensagens, biblioteca de classes incompletas, homem do meio, intimidade inapropriada e inveja de recursos**. Como uma boa possível opção de refatoração tem-se Substituir método para método objeto.

Ausência de duplicidade:

Um bom código é um código que não contém duplicidade, a duplicidade de código causa uma redundância desnecessária , implica em um código mais sujo e impacta diretamente na simplicidade e na elegância de um código. Sendo assim deve-se evitar ao máximo códigos duplicados.

Mau Cheiro :

A duplicidade de código por si só já é um mal cheiro de código,tendo como uma boa possível opção de refatoração é a **extração de método**.

Extensibilidade:

A extensibilidade de um código é um dos pontos principais para a sua escalabilidade, realizar a escrita de um código já imaginando a adição de uma nova funcionalidade facilita a escalabilidade de todo o projeto.Porém esse mesmo ponto pode acabar transformando o seu programa em um sistema operacional completo a depender da quantidade de “ganchos “ para inclusão de features que forem adicionadas no projeto. Nesse ponto é necessário um cuidado extra dos desenvolvedores para equilibrar as funcionalidades necessárias para a função no momento, o que definitivamente será utilizado e desenvolvido posteriormente, é oque pode ser necessário, para assim determinar o quão extensiva deve ser o seu software.

Mau Cheiro :

Um mal cheiro que pode ser identificado e uma **generalidade especulativa**, indicando que não foi realizado o equilíbrio ideal. Como uma boa possível opção de refatoração tem-se Colapsar hierarquia.

Boa documentação:

A documentação tem o intuito de ajudar os leitores do seu código a entender seu funcionamento, evitando ter que decifrar sua estrutura. Essa característica é especialmente importante para projetos mais complexos. A documentação deve ser pequena pois o projeto deve ser simples. Pode-se documentar o projeto escrevendo seu código de forma que ele seja facilmente compreensível, com diagramas e texto ou com programação literal para, por exemplo, uma API. A documentação beneficia seu projeto deixando seu conteúdo mais claro para desenvolvimento posterior.

Mau Cheiro :

Pode-se associar o mau cheiro **comentários** à essa característica, em virtude de que com boa documentação os comentários não são necessários. Como uma boa possível opção de refatoração tem-se Substituir parâmetros com métodos explícitos.

Referencia Bibliografica:

-Goodliffe, Pete. *Code Craft : The Practice of Writing Excellent Code*, No Starch Press, Incorporated, 2006. *ProQuest Ebook Central*,