

DOCUMENTAÇÃO DELIVERY LANCHE MASTER

Refatoração de Código Legado



(Fonte da Imagem: [Dribble](#))

Alice Braga da Silva

14.02.2022

Extensão JAVA - SoulCode

INTRODUÇÃO

Durante as aulas de Extensão de Java, sentiu-se a necessidade de lidar com códigos já existentes. Para criar hábito e experiência com esse tipo de projeto, foi pedido em uma atividade a refatoração de um código legado.

O projeto consiste em um Delivery de uma Lanchonete onde é possível escolher entre um menu de opções o Lanche que o usuário deseja. Porém, observou-se que a estrutura do código estava confusa e repetitiva.

OBJETIVO GERAL

Refatorar o código.

OBJETIVOS ESPECÍFICOS

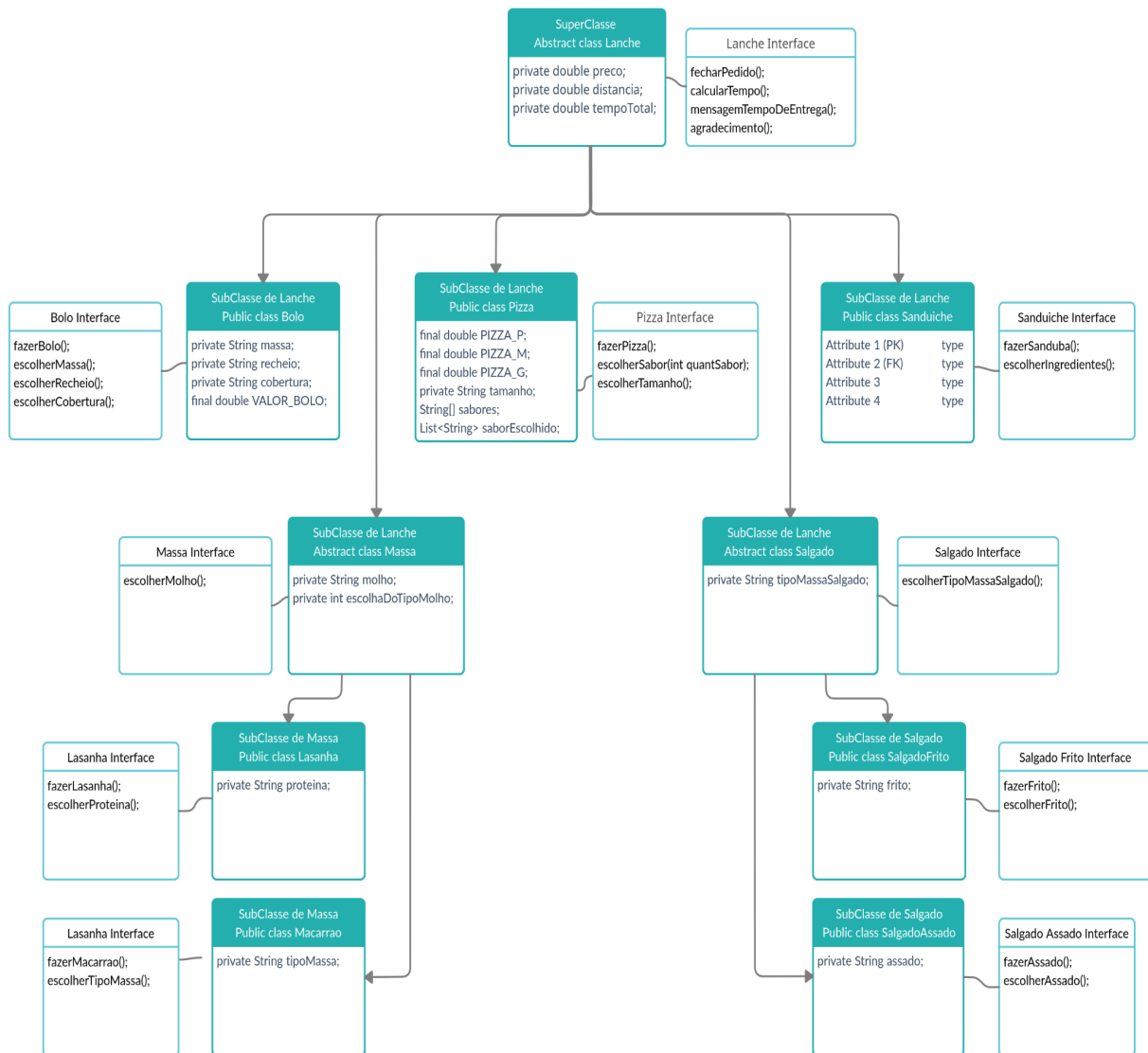
1. Analisar, entender e comentar o código.
2. Identificar falhas e corrigir, implementando melhorias de usabilidade, agradabilidade e estruturação de código.
3. Implementar uma quarta opção de compra: "Salgados", que devem ser divididos entre as opções de fritos ou assados. Para cada uma dessas opções, oferecemos 4 tipos de salgados.
4. Construir uma documentação para a aplicação.

DESENVOLVIMENTO

Ao analisar o código, conclui-se que: a estrutura da classe Main estava muito poluída, funções comuns para todas as subclasses estavam descentralizadas (ou seja, sendo repetidas ao longo do código sem necessidade), erros de ortografia, má utilização do conceito de herança e polimorfismo e alguns atributos comuns estavam repetidos.

1. Refatorar a classe Main, distribuindo os métodos para a superClasse Lanche;
2. Identificar os métodos e atributos repetidos para limpar as subClasses;
3. Inserir Interfaces para cada classe.
4. Mudar a Pizza para subClasse de Lanche e não como subClasse de Massa;
5. Criação da classe Salgado e suas subClasses Salgado Assado e Salgado Frito.

Para melhor visualizar o fluxo de informações e auxiliar na refatoração do código, um esquema foi desenvolvido pela plataforma *Creately.com* e será apresentado na imagem a seguir. É possível detectar as relações entre as classes e suas respectivas interfaces com seus métodos exclusivos.



(Fonte da Imagem: Alice Braga, desenvolvido em Creately.com)

RESULTADO

O fluxo dos códigos ficaram mais entendíveis e centralizados. Os atributos melhor utilizados pois seguiram as práticas dos modificadores de acesso. As lógicas ficaram mais enxutas pois em alguns casos foram utilizadas as funções de ArrayLists. As Interfaces auxiliaram na implementação de métodos. Para aplicar e praticar os conceitos, a criação da nova opção do menu (Salgado) que é dividido entre Salgado Assado e Salgado Frito foi feito usando os conceitos de herança, onde o Salgado é uma classe abstrata que possui um método em comum com as suas subClasses SalgadoAssado e SalgadoFrito.

CONCLUSÃO

Essa atividade foi necessária e produtiva, tanto para a prática de identificação e manutenção de códigos legados, quanto para colocar em prática os conceitos e conteúdos vistos em sala de aula. Também, notou-se a diferença entre códigos que possuem comentários, sejam eles para identificar as funções ou para explicar as lógicas desenvolvidas.