2018/1

UFMG

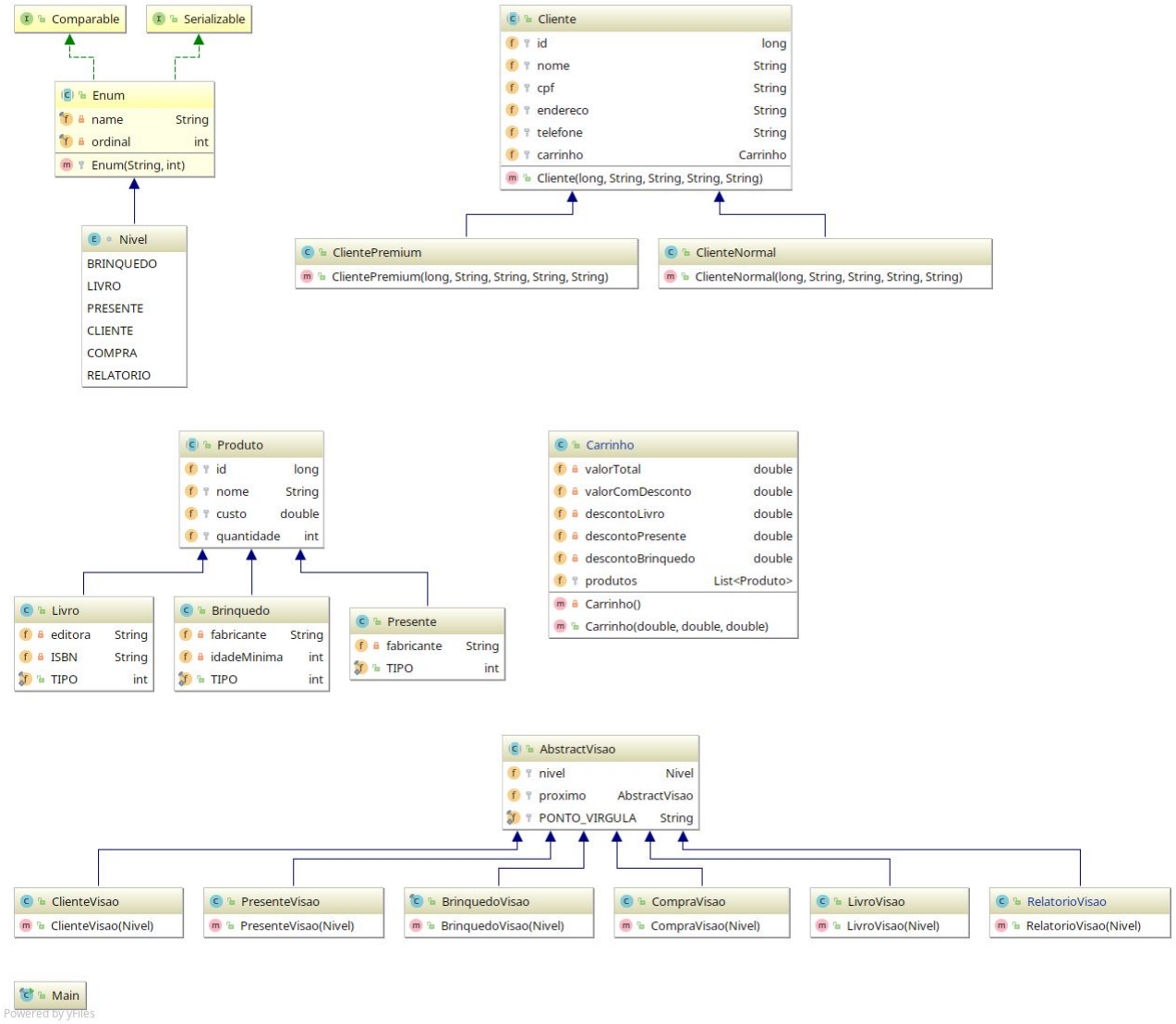
DCC052 - Programação Modular

Relatório Code Challenge

7 de maio de 2018

Antônio Côrtes
Paula Jeniffer
Rafael Perez
Sonia Fraga

Diagrama de classes



Decisões e Padrões de projeto

Foi utilizado o modelo de MVC, e separamos o modelo da visão. Tratando-se de padrões intrínsecos ao Java, usamos CamelCase para os nomes, seguindo o padrão da linguagem.

Ressalta-se que a ID dos produtos no arquivo compras não está compatível com as IDs dos produtos nos seus respectivos arquivos. O grupo entrou em acordo e tomou a decisão de usar a operação mod na lista de todos os produtos para resolver este problema, de modo a tratar as IDs apresentadas como um hash.

Desconhecemos se ocorreu algum erro na hora de criar os arquivos, então foi escolhida esta abordagem para manter o exemplo utilizável. Esta abordagem foi usada na função removerProdutoPorId, no arquivo Visao.java.

Padrão comportamental: Chain of Responsability

Este padrão foi usado para carregar os dados a partir da leitura de arquivos que, conforme especificado, é feita de modo sequencial.

Desta forma, o uso deste padrão fez-se apropriado devido a haver uma responsabilidade superior - uma visão centralizadora - que conhece a perspectiva geral das responsabilidades: a classe Main. Nela há o método getCadeiaDeVisoes, que realiza o encadeamento de todas as visões implementadas - visões estas independentes entre si. Assim atinge-se o desacoplamento entre estas visões.

Template

Não foi usado sobrescrita diretamente nas classes, porém a definição da herança nas classes abstratas permitiu simplificação das estruturas.

Polimorfismo

Foram utilizados dois tipo de polimorfismo: Coerção e Inclusão. Este primeiro foi utilizado para diferenciar o preço dos produtos, que pode variar de um brinquedo, um presente ou um livro, dependendo do tipo de produto desejado para o cálculo.

Já a Inclusão foi utilizada no momento da modelagem do tipo Produto que gera subtipos Brinquedo, Livro e Presente. Para Cliente o mesmo, que no caso possui duas subclasses: ClienteNormal e ClientePremium.

Pacotes

Com relação aos pacotes, para uma melhor organização e para melhor compreensão do código, as classes foram separadas de acordo com suas responsabilidades em comum. Os pacotes criados são:

- loja: possui as classes Produto, Brinquedo, Presente e Livro;
- usuario: possui as classes Cliente, ClientePremium e ClienteNormal;
- controle: possui a classe Carrinho;

Referências

- https://en.wikipedia.org/wiki/Naming_convention_(programming)#Java
- https://en.wikipedia.org/wiki/Chain-of-responsibility_pattern