

Projeto da disciplina Laboratório de Programação II

31 de Julho de 2018

O que foi realizado

Na US1 foi criada a classe Item, uma classe abstrata que representa todos os itens do sistema. E então, criada mais três classes herdando da classe Item: uma que representa os itens com quantidade fixa, uma que representa os itens por quilo e uma que representa os itens por unidade. Foi escolhido esse design, pois os itens possuem atributos e métodos em comum, por isso achamos melhor abstrair a classe Item. Criamos uma classe ControllerItem que é responsável por armazenar, os itens cadastrados no sistema os métodos para criar, atualizar, exibir e deletar itens do sistema. Os itens são armazenados em um TreeMap, pois cada item tem um id para ser identificado. Além disso, foram criados os testes de unidades para as classes com lógicas testáveis.

Na US2 foram criadas classes que representam controladores para fazer ordenação dos itens cadastrados. Foi criado a classe OrdenarItensPorNome para ordenar itens pelo nome, foi criada a classe OrdenarPorCategoria para ordenar os itens pela categoria, foi criada a classe OrdenarItensPorMenorPreço para ordenar itens pelo menor preço e foram criados os métodos no controlador de itens para realizar essas funcionalidades. E também foi feito o método para buscar um item a partir do seu nome(fazendo a busca nos itens cadastrados) e a resposta da busca será em ordem alfabética. Além disso, foram criados os testes de unidades para esses métodos.

Na US3 foi criada a classe Lista De Compras que representa uma lista de compras, que tem um TreeMap que associa um Item a uma classe Compra. Essa classe Compra guarda o Item e a quantidade que pretende ser comprada naquela lista de compras que ele está contido e essa classe agora também é responsável por atualizar a quantidade de um item em uma lista de compras. Foi criado um controlador para lista de compras que possui um TreeMap que associa o descritor da lista à ListaDeCompras, e é responsável por criar uma lista, adicionar itens a essa lista, atualizar a quantidade desse item em uma data lista, deletar um item de uma lista e finalizar uma lista de compras. Na exibição dos itens que estão em uma lista de compras são ordenados em duas camadas, pela categoria e em caso de empate pelo nome, para resolver o problema fizemos a ordenação pelo nome e então pela categoria já que o Java não anula a ordenação já feita antes. A classe ControllerDeListasDeCompras tem acesso a todos os itens cadastrados, pois recebe como parâmetro no construtor o controlador de itens.

Na US4 foram criados métodos para pesquisar listas de compras pelo descritor, pela data e por Item, o método pesquisaListaDeCompras, pesquisa e retorna o descritor da lista de

compras com o descritor passado como parâmetro, já o método `pesquisaListasDeComprasPorData` pesquisa e retorna os descritores de todas as listas de compras criadas na data passada como parâmetro e o método `pesquisaListasDeComprasPorItem` que pesquisa e retorna a data de criação e o descritor de todas as listas de compras que possuem o item passado como parâmetro.

Na US5 foram implementados os métodos no `ControllerListaDeCompras` para a geração automática das listas de compra. Sendo divididos por estratégias, sendo elas: `geraAutomaticaUltimaLista`, que apenas copia a última lista adicionada ao sistema, `geraAutomaticaItem`, que copia a última lista em que o item passado como parâmetro está presente, `geraAutomaticaItensMaisPresentes`, que gera uma nova lista contendo os itens mais cadastrados nas listas de compra do sistema.

Na US6 foi criada a classe `Supermercado`, para o melhor gerenciamento dos estabelecimentos de compra. Também foi criado o método `sugereMelhorEstabelecimento` que ordena os Supermercados de acordo com o valor total de determinados itens e a partir da posição passada como parâmetro é retornado o `toString` do respectivo Supermercado. A partir do mesmo método também é possível imprimir o `toString` de um item que esta dentro de um Supermercado.

Na US7 foram criados os métodos de `salvarDados` e `carregarDados` nos respectivos `Controllers` para que as informações dos itens e das listas de compra presentes no software permaneçam ativos mesmo após o mesmo ser finalizado e reiniciado.

Links úteis

Diagrama:

<https://drive.google.com/file/d/1nEb6keHoAH5x4pthC9lGggCncBpPKWak/view?usp=sharing>

GitHub:

https://github.com/IgorSilveira7/Projeto_LP2/