

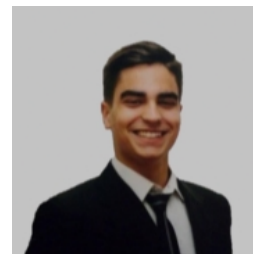
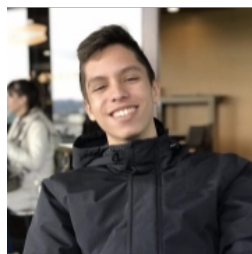


Universidade do Minho
Escola de Engenharia

Programação Orientada a Objetos

Grupo 40

Hugo Filipe Silva Abelheira - A95151
João Miguel Rodrigues da Cunha - A96386
Afonso Xavier Cardoso Marques - A94940



11 de maio de 2023

Conteúdo

1	Introdução	2
2	A aplicação desenvolvida	2
2.1	Classes	2
2.1.1	Main	2
2.1.2	MenuUI	2
2.1.3	Utilizador	2
2.1.4	Artigo	2
2.1.5	Sapatilha	3
2.1.6	Mala	3
2.1.7	T-Shirt	3
2.1.8	Fatura	4
2.1.9	Transportadora	4
2.1.10	Encomenda	4
2.1.11	Vintage	5
3	Problemas e soluções	5
4	Manual de Instruções	5
5	Conclusão	7
6	Anexos	7

Lista de Figuras

1	Menu de carregamento	8
2	Menu de autenticação	8
3	Menu principal	9
4	Menu de queries	9
5	Vista da loja	10

1 Introdução

No âmbito do trabalho prático da disciplina Programação Orientada a Objetos, foi proposto pela equipa docente o desenvolvimento de um sistema *marketplace*, que permite a compra e venda de artigos novos ou usados. Este relatório visa ilustrar o processo de desenvolvimento realizado pelo nosso grupo.

Iremos explorar as classes que foram criadas, o funcionamento geral do programa e as dificuldades sentidas durante o desenvolvimento. Será também apresentado um diagrama de classes bem como um manual de utilização do sistema, para auxiliar os utilizadores menos experientes.

2 A aplicação desenvolvida

A aplicação consiste num sistema de marketplace chamado Vintage que permite a compra e venda de artigos novos e usados de três tipos - sapatilhas, malas e t-shirts. O sistema tem como base os seus utilizadores, que atuam como compradores e vendedores, e têm a possibilidade de fazer encomendas de vários artigos de vendedores diferentes.

É também de notar que o utilizador tem acesso a uma panóplia de opções que permitem efetuar estatísticas sobre o estado atual do sistema bem como guardar e carregar o estado para um ficheiro binário à sua escolha.

De seguida expomos as várias classes implementadas no projeto desenvolvido

2.1 Classes

2.1.1 Main

Classe responsável por arrancar com o programa. Para isto o método `main` inicia a execução da classe `MenuUI`.

2.1.2 MenuUI

Classe que gere o fluxo da aplicação. Esta é a classe responsável por gerir o menu inicial e respetivos submenus que interage apenas com uma única classe, a `Vintage`. É aqui que lidamos com todos os inputs do utilizador ativo.

2.1.3 Utilizador

Classe que representa um utilizador. No contexto deste trabalho, consideramos que um utilizador pode ser comprador e vendedor simultaneamente. Os seus atributos consistem num código único atribuído pelo sistema, o seu endereço de correio eletrónico, a sua password, nome, morada, número fiscal e três listas que representam o inventário pessoal, a primeira com os artigos para venda, a segunda com os artigos comprados e a terceira com os artigos vendidos.

Existem aqui métodos que permitem consultar a informação do utilizador bem como fazer a gestão do inventário através da adição e remoção de artigos das três listas referidas anteriormente.

2.1.4 Artigo

Classe abstrata que é responsável por representar um artigo no sentido mais genérico possível tendo em conta o contexto do trabalho. Possui um código alfanumérico único atribuído pelo sistema, um estado que pode ser "novo" ou "usado", uma descrição atribuída pelo vendedor, a marca, o preço, a quantidade de donos anteriores (se for usado) e a transportadora que o vendedor escolheu.

Tem três subclasses, nomeadamente a `Sapatilha`, a `Mala` e a `T-Shirt`. Seguem-se as suas descrições.

2.1.5 Sapatilha

Uma das subclasses de Artigo, a classe Sapatilha herda os mesmos atributos da sua superclasse mas não deixa de possuir os seus atributos específicos, nomeadamente o tamanho de calçado, a forma como aperta (atrilhos ou atacadores), a cor e o ano de coleção.

O cálculo do preço final das sapatilhas é feito com base no preço e desconto estabelecido pelo vendedor, no preço de expedição da transportadora que foi associada, no tamanho e se são novas ou usadas. Seguem-se as formulas usadas:

- Sapatilhas novas com tamanho inferior ou igual a 45:
 $\text{preçoFinal} = \text{preçoVendedor} + \text{valorExpedição};$
- Sapatilhas novas com tamanho superior a 45:
 $\text{preçoFinal} = \text{preçoVendedor} - (\text{preçoVendedor} * \text{desconto}) + \text{valorExpedição};$
- Sapatilhas usadas:
 $\text{preçoFinal} = \text{preçoVendedor} - (\text{preçoVendedor} / \text{númeroDonos} * \text{desconto}) + \text{valorExpedição};$

2.1.6 Mala

Uma das subclasses de Artigo, a classe Mala, por herança, tem os mesmos atributos da sua superclasse. Possui os seus atributos específicos, nomeadamente as dimensões (altura, largura e profundidade), o material do qual é feita e o ano de coleção.

O cálculo do preço final das malas é feito com base no preço estabelecido pelo vendedor, no preço de expedição da transportadora que foi associada ao artigo e se este é novo ou usado. O valor do desconto é proporcionalmente inverso ao volume. Seguem-se as formulas usadas:

- Mala nova:
 $\text{preçoFinal} = \text{preçoVendedor} + \text{valorExpedição};$
- Mala usada:
 $\text{preçoFinal} = \text{preçoVendedor} - (\text{preçoVendedor} / \text{númeroDonos} * \text{desconto}) + \text{valorExpedição};$
- Cálculo do desconto com base no volume:
 $\text{desconto} = 0.1 * \exp(-0.0001 * \text{volumeMala});$

2.1.7 T-Shirt

A mais simples de entre as subclasses de Artigo, a classe T-Shirt possui os mesmos atributos de Artigo juntamente com os seus atributos específicos, nomeadamente o tamanho (S, M, L, XL) e o padrão, que pode ser liso, às riscas ou com desenhos de palmeiras.

O cálculo do preço final da t-shirt é feito com base no preço estabelecido pelo vendedor, no preço de expedição da transportadora que foi associada ao artigo, no facto de ser nova ou usada e no padrão. Seguem-se as fórmulas usadas:

- T-Shirt nova:
 $\text{preçoFinal} = \text{preçoVendedor} + \text{valorExpedição};$
- T-Shirt usada com padrão liso (não tem desconto):
 $\text{preçoFinal} = \text{preçoVendedor} + \text{valorExpedição};$
- T-Shirt usada com padrão às riscas ou palmeiras (desconto fixo de 50%):
 $\text{preçoFinal} = \text{preçoVendedor} - (\text{preçoVendedor} / \text{númeroDonos} * 0.5) + \text{valorExpedição};$

2.1.8 Fatura

A classe Fatura, no contexto do nosso trabalho, serve para criar um registo da venda de cada artigo da encomenda que foi feita por um utilizador. Esta decisão foi tomada pelo grupo com o propósito de haver uma melhor organização da informação e para se tornar mais acessível o processo de efetuar as estatísticas sobre o sistema bem como o ato de efetuar uma devolução de encomenda.

Cada fatura vai apresentar a informação não só da encomenda que foi feita mas também a data em que foi efetuada, número da encomenda, nome e NIF do utilizador que vendeu o artigo e do utilizador que fez a compra.

Para além disso, também podemos encontrar na fatura o código alfanumérico do artigo e o seu preço bem como a transportadora e o preço da expedição.

2.1.9 Transportadora

Esta classe tem o propósito de criar as transportadoras que vão distribuir os produtos após uma encomenda ser feita. Os objetos desta classe são definidos pelo seu nome, a sua margem de lucro e o seu imposto.

Cada Artigo que existe no sistema tem uma Transportadora atribuída pelo vendedor, o que vai afetar o preço final do artigo pago pelo cliente. O valor de expedição pode ser calculado da seguinte maneira:

- $\text{valorExpedição} = \text{margemLucro} * (1 + \text{imposto}) * 0.9;$

2.1.10 Encomenda

Esta classe representa cada encomenda que é feita por um utilizador. No contexto do trabalho, atribuímos três diferentes estados quando uma encomenda for feita, pode estar pendente, finalizada e expedida. Para alterar o estado de uma encomenda de finalizada para expedida usamos um método que compara a data da encomenda com a data do sistema e averigua se já se passaram 48 horas (dois dias). Se o resultado for positivo, o estado da encomenda é alterado e os artigos são colocados na lista de artigos comprados do cliente.

Também foi implementado no nosso trabalho a possibilidade de um Utilizador devolver uma encomenda, podendo também ser considerado de certa forma um estado. Para um utilizador devolver uma encomenda, é imperativo que a encomenda tenha sido expedida em primeiro lugar (o utilizador não vai devolver artigos que ainda não lhe foram entregues) e existe um período de 48 horas para o fazer. Se estas condições não se verificarem, o sistema não permite realizar a devolução.

Para além disso, cada encomenda pode incluir um ou mais artigos, e a quantidade de artigos comprados vai influenciar o valor base de cada uma. De seguida, apresentamos os valores base definidos pelo grupo:

- Caso uma encomenda tenha apenas 1 artigo - $\text{valorBase} = 1.99\text{€};$
- Caso uma encomenda tenha entre 2 e 5 artigos - $\text{valorBase} = 2.99\text{€};$
- Caso uma encomenda tenha mais que 5 artigos - $\text{valorBase} = 3.5\text{€};$

2.1.11 Vintage

Classe responsável pela gestão de todas as entidades existentes na Vintage sendo por isso a de maior dimensão. Possui a informação de todos os artigos, utilizadores, encomendas, transportadoras e faturas sobre o qual as interações vão ocorrer. Os objetos mencionados são guardados em estruturas de Map. Existem ainda outras duas variáveis: o *vintageBank* que representa o dinheiro que a Vintage possui e o *user_atual* que é um objeto da classe Utilizador que representa o utilizador que está atualmente autenticado no sistema.

Nesta classe existem muitos métodos. Alguns exemplos incluem os métodos que lidam com os inputs dados no MenuUI, como são os casos dos métodos usados para fazer uma encomenda, criar novos artigos, criar novas transportadoras, fazer devoluções de encomendas e responder às queries e estatísticas. É também aqui que se encontram os métodos que permitem guardar e carregar o ficheiro de estado no programa.

3 Problemas e soluções

No decorrer do desenvolvimento do nosso sistema, o grupo deparou-se com poucas, mas cruciais dificuldades. Quando começámos a tratar da funcionalidade que permite ao utilizador devolver encomendas, percebemos que embora fosse muito fácil cancelar a encomenda, pois, devido à nossa organização, seria apenas necessário remover da estrutura que contém todas as encomendas, voltar a colocar os artigos da encomenda nos respetivos *maps* que contém os artigos para venda de um utilizador, ia ser muito desafiante, devido ao facto de perdermos a informação relativa aos vendedores.

Para ultrapassar este problema, decidimos guardar toda a informação deste procedimento, isto é, as informações do comprador, do vendedor e da encomenda, nas faturas. Agora, podemos simplesmente iterar sobre este *map*, recolhendo e guardando a informação necessária para obter os identificadores únicos de cada vendedor e comprador, permitindo que possamos aceder às respetivas estruturas de dados para posterior adição ou remoção do artigo/encomenda.

É também importante mencionar que efetivamente não conseguimos implementar nenhuma arquitetura para organizar as classes do nosso sistema. Inicialmente começamos apenas por nos preocupar com o encapsulamento das nossas classes, portanto decidimos fazer a gestão dos dados todos na classe *Vintage*. Mais tarde, com o decorrer do semestre, percebemos que seria extremamente útil para o nosso projeto fazer a divisão em componentes, onde cada um iria conter uma classe com a respetiva *interface* e classe *facade*. No entanto, o tempo acabou por ser um fator limitador para implementar esta arquitetura, e acabamos por manter o nosso plano inicial, implementando assim uma arquitetura de MVC extremamente rudimentar, onde o único controller do nosso projeto é, efetivamente, a classe *Vintage*.

4 Manual de Instruções

Segue-se então um manual para poder utilizar o nosso sistema. É de notar que a nossa interface consiste num menu de texto básico que lê e processa valores de input dados pelo Utilizador.

Existem três menus e dois submenus. Ao iniciar o programa é apresentado ao utilizador o primeiro menu, chamemos-lhe menu de carregamento, que dá a opção de iniciar o sistema com ou sem estado. Se o utilizador opta por iniciar o sistema sem estado, terá de indicar qual é o ficheiro onde vai querer guardar posteriormente e terá de fazer o povoamento com novas entidades. A outra opção permite indicar qual é o ficheiro binário que se vai carregar no sistema. Tendo feito este procedimento avança-se para o menu de autenticação.

No menu de autenticação o objetivo é claro: autenticar uma conta existente ou criar uma nova conta. O ato de registar uma nova conta resulta na criação de um novo objeto da classe Utilizador que é de seguida adicionado ao sistema.

De seguida segue-se o menu que consideramos ser o "principal". Aqui encontram-se as opções iniciais:

Opções de utilizador:

- Entrar na loja - Opção que permite consultar todos os artigos de todos os vendedores bem como preencher o carrinho de compras para fazer uma encomenda;
- Adicionar artigo para venda - Opção que permite indicar o tipo de artigo (sapatilha, mala ou t-shirt) que o utilizador quer colocar para venda. De seguida introduz os atributos do artigo e este é adicionado ao seu catálogo pessoal.
- Remover artigo para venda - Esta opção permite remover um artigo que o utilizador tenha para venda. Precisa apenas de indicar o código único.
- Listar os artigos que tenho para venda - Permite visualizar os artigos que o utilizador tem disponíveis para vender.
- Registo de todos os artigos que comprei - Permite visualizar os artigos que o utilizador comprou. Se o utilizador devolve uma encomenda, os artigos que lá estavam incluídos são removidos desta lista.
- Registo de todos os artigos que já vendi - Permite visualizar os artigos que o utilizador vendeu. Se um cliente devolve uma encomenda, os artigos que originalmente eram do vendedor são removidos da sua lista de artigos vendidos e retornam à sua lista de artigos para venda.
- A minha conta - Permite visualizar a informação do utilizador atual.
- Fazer devolução de encomenda - O utilizador indica o número único de encomenda e, se todas as condições estiverem reunidas, esta será devolvida.

Opções de sistema:

- Criar nova transportadora - Permite criar um novo objeto da classe Transportadora para ser adicionado ao sistema.
- Queries ao sistema - Escolhendo esta opção, o utilizador é encaminhado para um submenu com oito opções extra que lhe permitem fazer questões estatísticas sobre o estado atual do sistema. Algumas dessas opções incluem determinar qual foi o vendedor que mais faturou num dado período de tempo, saber qual foi a transportadora que mais lucrou, quanto ganhou a Vintage durante o seu funcionamento e a listagem de faturas, encomendas, e artigos existentes no sistema.
- Guardar estado - Opção que permite guardar o estado do sistema num ficheiro binário. O ficheiro onde se irá guardar é aquele que foi indicado no menu de carregamento logo no início da execução do programa.
- Mudar data/hora - Opção que permite mudar a data do sistema. Só é possível andar para a frente no tempo.

Implementámos ainda uma opção no menu principal que nos dá acesso a várias *queries*, isto para o utilizador ter uma melhor percepção daquilo que se está a passar no sistema, bem como ter acesso a algumas estatísticas que podem vir a ser úteis.

Segue-se a explicação de cada funcionalidade do submenu das *queries*:

- Qual o vendedor que mais faturou - O sistema imprime no ecran qual o vendedor que mais lucrou na Vintage durante um certo período de tempo. Se quiser, o utilizador tem a possibilidade de estabelecer um intervalo de tempo personalizado para a pesquisa.
- Qual a transportadora com maior volume de facturação - Semelhante à query anterior, o sistema imprime no ecran qual a transportadora que mais lucrou na Vintage durante um certo período de tempo. Aqui o utilizador também pode estabelecer um intervalo de tempo personalizado para a pesquisa.
- Listar as encomendas do sistema - Permite visualizar todas as encomendas que foram efetuadas durante o decorrer do programa.
- Listar as faturas do sistema - Semelhante à opção anterior, permite visualizar todas as faturas efetuadas durante o decorrer do programa.
- Ordenar os maiores vendedores do sistema durante um certo período - Aqui o utilizador introduz duas datas e o programa devolve uma lista ordenada de forma decrescente, dos vendedores em relação ao lucro.
- Ordenar os maiores compradores do sistema durante um certo período - Aqui o utilizador introduz duas datas e o programa devolve uma lista ordenada de forma decrescente, dos compradores em relação ao lucro.
- Quanto dinheiro ganhou o Vintage no seu funcionamento? - Permite saber a quantidade exata de lucro do nosso sistema Vintage, desde que foi inicializada pela primeira vez.
- Listar todos os artigos que passaram pelo sistema - Permite visualizar uma listagem de todos os artigos que alguma vez passaram pela Vintage, incluindo os artigos que foram removidos por um utilizador.

NOTA: Sempre que o utilizador tem que introduzir um valor de input que não seja apenas a opção que queira seleccionar, por exemplo, quando quer realizar uma encomenda, o sistema está preparado para imprimir no ecran mensagens de texto que visam auxiliar o utilizador, indicando-lhe o input correto.

5 Conclusão

No decorrer do desenvolvimento deste trabalho foi possível aplicar conhecimentos adquiridos nas aulas da Unidade Curricular de Programação Orientada aos Objetos, bem como a utilização da linguagem de programação Java.

Salientamos como ponto positivo a quantidade de funcionalidades que conseguimos implementar, o esforço feito para manter o encapsulamento e a arquitetura escolhida que permite uma separação clara entre a camada de interface e a camada de negócio do programa.

No entanto, fazendo uma reflexão crítica sobre o nosso trabalho, somos capazes de admitir que a arquitetura podia estar melhor, implementando, por exemplo, uma estrutura complexa de *MVC* (*Model View Controller*) ou usando classes *Facade*.

6 Anexos


```
----- Vintage v.1 -----  
  
-----  
1) Iniciar Vintage sem estado  
2) Carregar ficheiro de estado  
0) SAIR  
-----  
  
2  
Insira o nome do ficheiro de estado...  
VintageData
```

Figura 1: Menu de carregamento

```
-----  
1) Autenticar conta  
2) Registrar nova conta  
0) SAIR  
-----  
  
1  
Insira o seu email...  
afonso@mail.com  
Insira a sua password...  
afonso_pass
```

Figura 2: Menu de autenticação

```
----- TIME:2023-05-10 --  
---OPÇÕES DE UTILIZADOR---  
1) Entrar na loja  
2) Adicionar artigo para venda  
3) Remover artigo para venda  
4) Listar os artigos que tenho para venda  
5) Registo de todos os artigos que comprei  
6) Registo de todos os artigos que já vendi  
7) A minha conta  
8) Fazer devolução de encomenda  
  
---OPÇÕES DE SISTEMA---  
9) Criar nova transportadora  
10) Queries ao sistema  
11) Guardar estado  
12) Mudar data/hora  
0) SAIR  
-----
```

Figura 3: Menu principal

```
-----  
1) Qual o vendedor que mais faturou...  
2) Qual a transportadora com maior volume de facturação...  
3) Listar as encomendas do sistema  
4) Listar as faturas do sistema  
5) Ordenar os maiores vendedores do sistema durante um certo período  
6) Ordenar os maiores compradores do sistema durante um certo período  
7) Quanto dinheiro ganhou o Vintage no seu funcionamento?  
8) Listar todos os artigos que passaram pelo sistema  
-----  
2
```

Figura 4: Menu de queries

Vendedor: 3 - Hugo									
ARTIGO (Sapatilha): Código: ART-6 Estado: usado Descrição: Sapatos usados Michael Kors Marca: Michael Kors Preço: 36.0€ Desconto: 0.23 Quantos donos já teve: 1 Tamanho: 38 Como se aperta: atacadores Cor: preto Ano de coleção: 2020									
ARTIGO (T-Shirt): Código: ART-7 Estado: novo Descrição: Camisola nova Marca: Guess Preço: 33.99€ Desconto: 0 Quantos donos já teve: 0 Tamanho: S Padrão: riscas									
ARTIGO (Mala): Código: ART-8 Estado: novo Descrição: Mala nova da Desigual Marca: Desigual Preço: 30.0€ Desconto: 0 Quantos donos já teve: 0 Dimensão: 18.0x22.0x10.0 Material: pele sintética Ano de coleção: 2023									
Vendedor: 4 - João									
ARTIGO (Sapatilha): Código: ART-9 Estado: usado Descrição: Botas de senhora usados Marca: Calvin Klein Preço: 37.0€ Desconto: 0.11 Quantos donos já teve: 1 Tamanho: 39 Como se aperta: atilhos Cor: preto Ano de coleção: 2022									
ARTIGO (T-Shirt): Código: ART-10 Estado: usado Descrição: Camisola com algum uso Marca: Boss Preço: 7.0€ Desconto: 0 Quantos donos já teve: 1 Tamanho: L Padrão: liso									
ARTIGO (Mala): Código: ART-11 Estado: usado Descrição: Mala com alguns anos Marca: Moschino Preço: 16.7€ Desconto: 0.01 Quantos donos já teve: 1 Dimensão: 30.0x40.0x20.0 Material: feltro Ano de coleção: 2019									
Vendedor: 1 - Carmelita									
ARTIGO (Sapatilha): Código: ART-0 Estado: novo Descrição: Sapatilhas muito bonitas Marca: Rebook Preço: 32.45€ Desconto: 0.3 Quantos donos já teve: 0 Tamanho: 43 Como se aperta: atacadores Cor: vermelho Ano de coleção: 2023									
ARTIGO (T-Shirt): Código: ART-1 Estado: usado Descrição: T-Shirt usada em bom estado Marca: Nike Preço: 10.5€ Desconto: 0.5 Quantos donos já teve: 1 Tamanho: S Padrão: palmérias									
ARTIGO (Mala): Código: ART-2 Estado: usado Descrição: Mala usada da gucci Marca: Gucci Preço: 20.3€ Desconto: 0.03 Quantos donos já teve: 1 Dimensão: 25.0x30.0x15.0 Material: couro Ano de coleção: 2022									
Vendedor: 5 - Emanuel									
ARTIGO (Sapatilha): Código: ART-12 Estado: usado Descrição: Sapatos em bom estado Marca: Calvin Klein Preço: 45.0€ Desconto: 0.35 Quantos donos já teve: 1 Tamanho: 41 Como se aperta: atilhos Cor: bege Ano de coleção: 2020									
ARTIGO (T-Shirt): Código: ART-13 Estado: usado Descrição: T-Shirt em segunda mão Marca: Macron Preço: 5.5€ Desconto: 0.5 Quantos donos já teve: 1 Tamanho: XL Padrão: riscas									
ARTIGO (Mala): Código: ART-14 Estado: usado Descrição: Mala em segunda mão Marca: Latouche Preço: 18.0€ Desconto: 0.05 Quantos donos já teve: 1 Dimensão: 20.0x25.0x15.0 Material: pele sintética Ano de coleção: 2017									

Escolher artigo introduza: 'num_vendedor, codigo_artigo'									
'Y' -> encomenda concluída									
'N' -> cancelar encomenda / sair da loja									
5,ART-13									
3,ART-6									

Figura 5: Vista da loja