**Utilizadores : (Henrique)**

1. Podem ser vendedores, compradores ou ambos.
2. Um vendedor pode publicar os seus artigos.
3. Um comprador pode comprar diversos artigos de uma vez(numa encomenda).
4. O vendedor é que escolhe a transportadora no momento da publicação do artigo, tendo todos os artigos para venda a sua transportadora definida.
5. O vendedor é que define os descontos.
6. Cada utilizador, deverá ter um código fornecido pelo Vintage, ao seu email (utilizado no login), nome, morada, numFiscal, e também o seu histórico de compras e vendas, e produtos que estão ainda à venda.

**Artigos : (Ema)**

1. Podem ser sapatilhas, t-shirts e malas(subclasses), podendo haver mais subclasses no futuro.
2. Cada artigo tem um código de barras único (alfanumérico).
3. Cada artigo pode ser novo ou usado, tendo no ultimo caso o seu estado e informação do numero de donos passados.
4. Tem também descrição, marca, preço base, e um desconto(que pode acontecer devido ao tipo de artigo em questão e do seu estado).
5. Estado de utilização varia entre 0 e 1.

**NOTAS: FALTA METER RESTRICOES NO PARAMETERIZADO CASO NAO SEJE NOVO E AINDA O ENUM. LIMITAR O COISO DE 0 A 1**

**Sapatilhas:**

1. Possuem tamanho numérico, tem atacadores/atilhos(um bool), cor e data de lançamento da sua coleção(1 por ano).
2. Apenas sapatilhas usadas podem ter desconto(que sejam de coleções de anos anteriores), ou sapatilhas novas de tamanho superior ao 45. Assumi 30%
3. precoBase = precoBase **-** (precoBase/numeroDonos ∗ estadoUtilizacao).
4. Existem sapatinhas **Premium**, onde a sua equação é precoBase = precoBase + (precoBase∗0.1\*nAnos).
5. Nessa caso, o nAnos será a data atual menos a data de lançamento da sua coleção.

**NOTAS: METER ENUM NAS CORES.VER SE O GET DESCONTO È PARA FICAR(Se não der para mudar tirar do equals). Confirmar se o desconto assim continua em composicao. Ver se meto o preco final ou não, se meter se meto na sub ou na outra. Mudar nome para data\_colecao**

**T-Shirts:**

1. possuem um tamanho (S,M,L,XL) e um padrao(liso, riscas e palmeiras).
2. As lisas não podem ter descontos.
3. As riscas e palmeiras tem 50% de desconto, se forem usadas.

**Tretas do costume dos enum e ver clones/agregacao**

**Malas:**

1. Possuem dimensão (pequeno(0.1), medio(0.25) e grande(0.5)), sendo que o desconto é o inverso do valor da sua dimensão (1/Vdimensão), o seu material e ano da coleção.
2. PrecoBase= PrecoBase – desconto.
3. Para as **Premium,** o preco aumenta segundo:

precoBase = precoBase + (precoBase∗0.1\*nAnos).

NOTAs: Meter enum nas dimensoes e coisas dos clones. Ver se as contas estao certas.

**Encomendas: (Flavio)**

1. Cada encomenda junta diversos artigos.
2. Cada encomenda, é cobrada uma taxa ao de garantia ao vendedor.
3. Cada encomenda poderá ter diversos artigos.
4. Cada encomenda tem 48h(após a entrega) para ser devolvida pelo comprador.
5. Possui um coleção (hashmap por ex) de artigos, a dimensão da embalagem (grande, pequeno, medio), preco final da encomenda(onde é somado uma taxa de 0.5 euros por cada artigo novo e de 0.25 euros por cada artigo usado e também os custos de expedição que variam entre transportadoras), e também um estado de encomenda.
6. O estado da encomenda é inicialmente pendente, sendo finalizada quando a mesma for paga. Depois, quando enviada passa para o estado expedida, tendo também no final a sua data de entrega.
7. Esta classe deve permitir criar encomendas, adicionar e remover artigos, calcular o preco final, e também a sua devolução caso esteja ainda dentro das 48h.

**Transportadoras: (Henrique)**

1. Transportam as diversas encomendas.
2. Cada transportadora tem o seu preço de expedição.
3. Cada transportadora define os seus preços, no entanto respeita as seguintes regras:

- Está definido no arranque do sistema qual é o valor base da expedição para encomendas pequenas (1 artigo), médias (2 a 5 artigos) e grandes (mais que 5 artigos).

- Está definido no arranque do sistema qual é o fator multiplicativo dos impostos, isto é, o sistema tem definido o valor do imposto a cobrar em cada transporte.

Ex: PrecoExpedicao = (ValorBase ∗ margemlucrotransportadora ∗ (1 + Imposto)) ∗ 0.9.

1. **OPCIONAL:** Um factor de valorização do sistema será a existência de transportadoras especializadas na entrega de artigos Premium, sendo que nem todas as transportadoras existentes no sistema cumprem este requisito. O preço de expedição para um artigo Premium pode seguir uma regra de cálculo diferente dos outros artigos, sendo que essa decisão cabe apenas à transportadora.
2. Se uma transportadora mudar a sua forma de calculo de preços, as que já estão em caminho não são afetadas.

**Vintage: (Todos)**

1. Faz o controlo dos stocks dos artigos, bem como das diversas encomendas feitas pelos compradores e vendas dos vendedores. **(Henrique)**
2. O sistema deverá ter a capacidade de avançar no tempo, sabendo também em que data se encontra.**(Flavio)**
3. Quando se avança no tempo, as encomendas que deveriam ser entregues nesse intervalo de tempo tem que estar entregues e os stocks do artigos diminuído. **(Henrique)**
4. O sistema deverá guardar a informação das diversas vendas, bem como o seu valor final.**(Ema)**
5. Quando uma encomenda é entregue, faz-se duas faturas para o vendedor e comprador com a informação da encomenda e preço final(get na encomenda). **(Ema)**
6. Metodos adicionais:

1. qual é o vendedor que mais facturou num período ou desde sempre **(Ema)**  
2. qual o transportador com maior volume de facturação **(Ema)**  
3. listar as encomendas emitidas por um vendedor **(Henrique)**  
4. fornecer uma ordenação dos maiores compradores/vendedores do sistema durante um período a deter-  
minar **(Flavio)**  
5. determinar quanto dinheiro ganhou o Vintage no seu funcionamento **(Henrique)**