

**Universidade de Caxias do Sul – Área de Conhecimento  
de Ciências Exatas e Engenharias**



**Avaliação de Disciplina**

**Código:** FBI4014

**Ano/** 2025/2

**Semestre:**

**Disciplina:** Programação Orientada a Objetos

**Turma:** AA

**Professor:**

**Data:** 24/04/2025

**Avaliação:** Trabalho de Implementação T1 e T2

**Leiam atentamente as instruções contidas nessa definição. O que será cobrado na avaliação está descrito aqui. Itens que faltarem causarão descontos na nota.**

**Objetivo :** Testar a proficiência dos alunos na implementação de um sistema utilizando os conceitos e práticas da Programação Orientada a Objetos.

**Forma de realização:** Os alunos se dividirão em grupos de até **3 elementos**, que realizarão a implementação de um sistema que será descrito a seguir. **Não serão aceitos trabalhos individuais!**

**Forma de Apresentação :** O trabalho deverá ser entregue em arquivo zipado, contendo o projeto Java na IDE Eclipse com todos os artefatos necessários para a execução do software, que deverá obrigatoriamente executar assim que importado no Eclipse do professor. **(Não serão aceitos trabalhos que não sejam enviados em formato Eclipse)**

**Data de Entrega :** O trabalho deverá ser entregue e apresentado em **22/05/2025 (parte 1)** e **03/07/2025 (parte 2)**. Após essas datas, os trabalhos perdem dois pontos por atraso (cada parte).

**Critérios de Avaliação :** Será realizada apresentação em sala de aula de cada trabalho, pelo respectivo grupo para o professor, e nessa apresentação os componentes do grupo terão que explicar como modelaram e implementaram partes da aplicação, conforme questionamentos do mesmo.

### **Serão avaliados:**

- A execução das tarefas descritas para o sistema (o sistema faz o que pede) ?
- A estruturação do modelo e subdivisão do sistema em classes (Obrigatório utilizar OO).
- A correção do código

### **O Projeto:**

Será implementado um sistema de pedidos de uma loja, compondo uma aplicação Desktop em modo caracter. O trabalho deverá obrigatoriamente utilizar a linguagem Java.

Como interface, será utilizado o modelo de menus em modo caracter apresentado em aula e fornecido juntamente com essa definição, no arquivo **ExemploMenu.zip**.

O modelo de negócio da loja está anexo neste documento.

Os alunos podem acrescentar classes e funcionalidades ao sistema como preferirem. NÃO é permitido excluir classes ou funcionalidades, sendo as citadas no modelo o mínimo aceitável.

Deverão ser implementados:

#### **Parte 1) (Ator Administrador)**

- Login / Exibição de menus conforme nível de acesso
- Cadastros de fornecedores
- Cadastro de produtos, relacionando-os aos fornecedores.
- Manutenção do estoque de produtos

Para cada cadastro, devem ser implementadas as seguintes funcionalidades:

- Inclusão
- Alteração
- Exclusão
- Consulta ( por código e por nome )

Na primeira parte dos trabalho, deverão ser utilizados vetores para conter os objetos. Na segunda, os vetores serão substituídos por Listas.

## **Parte 2)**

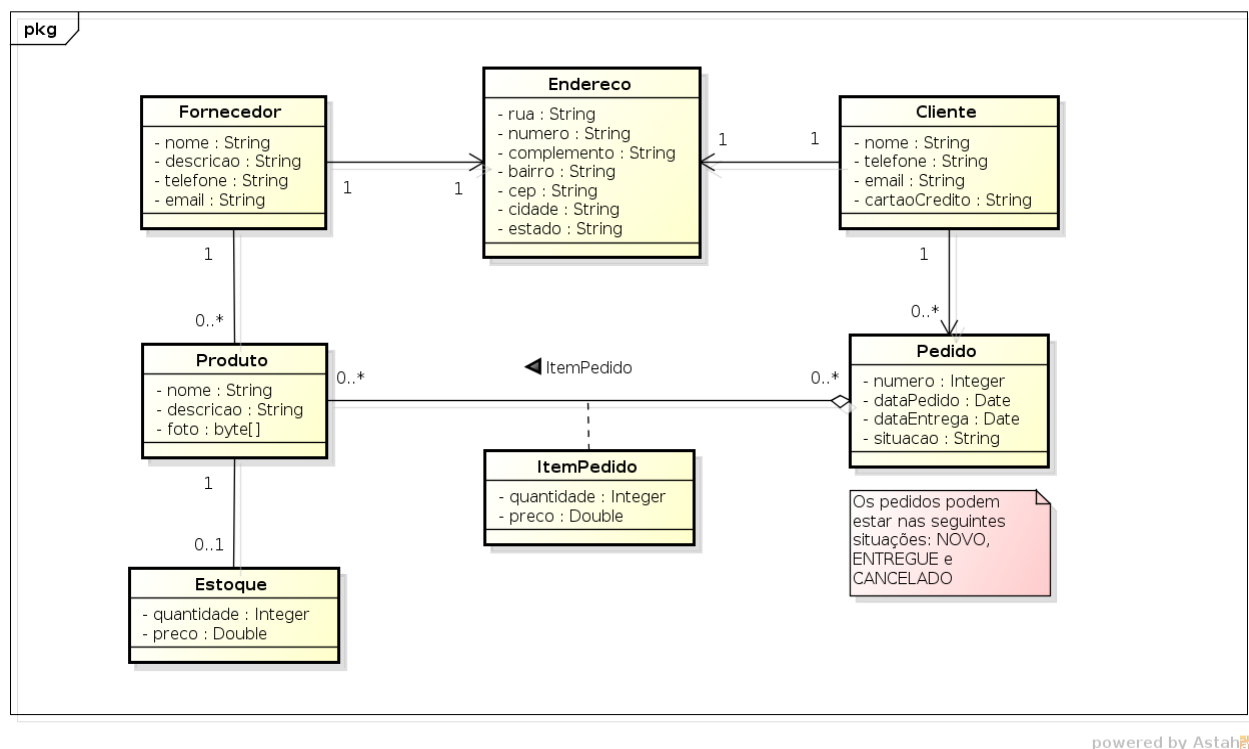
- Tela de consulta de produtos (com pesquisa) e realização de pedidos. (**Ator Cliente - Carrinho de compras.**)
  - O usuário poderá consultar em um menu específico para isso um produto qualquer, pesquisando pelo seu código ou por uma palavra contida em seu nome ou descrição. O sistema apresentará uma lista com os produtos que atendem essa pesquisa, o usuário então digita o identificador do produto, que pode ser o código ou a posição que o mesmo aparece na lista de resultados, e depois informa a quantidade de itens que deseja comprar. O sistema calcula e mostra quanto será o total do item e solicita a confirmação do usuário para adicionar o produto ao carrinho.
- Tela de consulta de pedidos por parte do cliente, onde será possível consultar um pedido feito por número do pedido ou os pedidos realizados em um determinado intervalo de datas. Para cada pedido, deverá ser possível visualizar seus detalhes, como quais foram os itens solicitados, o preço individual, quantidade e preço total de cada item. Lembrando que ao preço total de um pedido deve ser adicionado um valor de 17% de ICMS. Cada cliente só pode ver os seus pedidos.
- Regras para vendas:
  - Só se pode vender itens que estão em estoque
  - Itens com estoque ZERO devem ser mostrados como indisponíveis na consulta.
  - A realização do pedido deve diminuir a quantidade do estoque.
  - Se o cliente quiser comprar mais do que há em estoque deve receber uma mensagem de erro, oferecendo-lhe a quantidade máxima disponível.
  - Deve ser calculado o valor total do pedido e exibido ao cliente.
- Tela de consulta de pedidos realizados por parte da administração da loja ( Ator Interno) , onde será possível consultar e alterar o status do pedido, gravando para isso as datas de envio e / ou cancelamento.
  - Deve ser possível consultar pedidos pelo número do pedido ou pela data de realização do mesmo
  - Os pedidos devem ser exibidos uma tela em mestre-detalle, com o cabeçalho (mestre) mostrando os dados do pedido (data, nro, valor total, etc..) e o corpo

(detalhe) mostrando a foto, descrição, quantidade, valor unitário e valor total de cada item do pedido. Tanto usuários internos como o cliente que fez o pedido podem acessar esses dados.

**A aplicação deverá implementar todos os requisitos previstos neste documento.**

Como padrão de comportamento de aplicação, os alunos podem se basear em sites como Amazon, Mercado Livre ou AliExpress. O sistema será utilizado por dois atores: Internos e Clientes. Os usuários de sistema deverão acessar as funcionalidades do sistema através de menus, que deverão ser mostrados conforme o nível do usuário em questão (usuário interno ou externo), o que leva a necessidade de um cadastro de usuário / login, não demonstrado no diagrama. Os grupos devem modelar e implementar essa parte.

O modelo de classes da aplicação proposta é apresentado a seguir. Note-se que é possível evoluir o modelo da aplicação para um modelo que utilize herança de classes:



Deve ser observado que operações realizadas sobre as listas de objetos **devem ser feitas por referência**, e não por cópias de objetos. Se por acaso um produto que já estiver em um pedido de um cliente tiver sua descrição alterada, por exemplo, essa

alteração deve ser observada em todos os pedidos que possuírem aquele produto. Isso vale para todos os objetos.

O trabalho será dividido em duas partes. A primeira consiste-se da implementação das funcionalidades de Login, cadastros de Produtos (com Estoque), Clientes (com Endereço) e Fornecedores (com Endereço), entregue na data informada no início desta definição, e a segunda, com a aplicação completa, deverá ser entregue no final do semestre.

Na segunda parte do trabalho os dados deverão todos ser gravados em arquivos e recuperados a cada execução do programa, simulando uma aplicação que faça uso de bancos de dados. Como gravar os dados em arquivos será demonstrado em aula.

A segunda parte do trabalho também requer a utilização de exceções programadas pelo desenvolvedor, portanto, deve ser verificado no sistema onde podem ser colocadas tais exceções. Um exemplo é a quantidade insuficiente de produtos para realizar um pedido.

**Dúvidas perguntem ao professor!**

**Bom Trabalho!**