

# Trabalho de Orientação a Objetos

Professor: Gleiph Ghiotto Lima de Menezes

Autor: Augusto Benedito Carvalho da Silva - 202265552C

Link do repositório:

<https://github.com/Malungkot/Click-Go>

## 1. Objetivo

O objetivo deste trabalho é implementar um sistema de gerenciamento que englobe os princípios fundamentais da programação orientada a objetos, utilizando a linguagem Java para desenvolver o projeto e incorporando elementos essenciais como classes, pacotes, herança, polimorfismo, interfaces, coleções, tratamento de exceções, uma interface gráfica e leitura e gravação de arquivos.

## 2 - Domínio do aplicativo

Consiste no desenvolvimento de um marketplace autônomo, com recursos como registro de clientes, login de clientes e administradores. O recurso de administrador permite o gerenciamento de produtos, inclusive a adição ou remoção de produtos. Os principais requisitos desse sistema estão listados abaixo:

### Para o Cliente:

- Cadastro único por usuário, utilizando o CPF como "ID pessoal".
- Carrinho de compras, possibilitando o usuário comprar os produtos disponíveis no marketplace, como bebidas e alimentos;
- Possibilidade de pagamento de diferentes formas, alterando o valor final para o Cliente, com base na escolha do mesmo.

### Para o Administrador:

- Janelas separadas para controle dos Produtos e dos Administradores.
- Operações de Criar, Atualizar, Visualizar e Deletar para os Produtos e a Cadastrar administradores, como também remove-los.
- Registro em arquivos, que atuarão como banco de dados, que confere maior segurança e praticidade.

## 3 - Classes e Pacotes

### 1. Tratamento de Exceção

- **Uso:** Todas as classes de exceção personalizadas (como CampoVazio, CpfEmUso, etc.) são utilizadas para tratar erros de entrada de dados e situações anormais, garantindo que o sistema responda de maneira controlada a erros e mantenha a integridade dos dados.

### 2. Herança

- **Uso:** A classe Bebida é uma classe base da qual as classes BebidaComAlcool e BebidaSemAlcool herdam. Isso permite que métodos e atributos comuns sejam reutilizados, facilitando a manutenção e expansão do código.

### 3. Polimorfismo

- **Uso:** O polimorfismo é utilizado nas classes que implementam interfaces como ComAlcool, ComPeso, ComValor, e ComVolume. Isso permite que diferentes tipos de bebidas e alimentos sejam tratados de forma intercambiável, dependendo de suas implementações específicas.

### 4. Classes Abstratas e Interfaces

- **Uso:** As interfaces definem contratos para as classes, como as operações que devem ser implementadas por diferentes tipos de produtos (bebidas e alimentos). Isso ajuda a padronizar o comportamento esperado sem forçar uma implementação concreta.

### 5. Uso de Arquivos (Leitura e Escrita em Arquivos)

- **Uso:** O sistema pode incluir métodos para salvar e carregar informações de produtos e usuários a partir de arquivos, facilitando a persistência de dados entre sessões. Embora não tenha sido explicitamente detalhado, o método de BancoDeDados poderia incluir essa funcionalidade.

### 6. Interface Gráfica com o Usuário (GUI)

- **Uso:** O sistema utiliza a biblioteca Swing para criar interfaces gráficas, proporcionando uma experiência de usuário interativa. Classes como Cadastro, Login, dashAdmin, e dashCliente são exemplos de como a GUI é implementada para facilitar a interação do usuário com o sistema.

#### 1. Classes Model:

##### 1.1 Classe: Alimento

- **Descrição:** Representa os alimentos no sistema, armazenando informações como nome,

estoque, preço e peso.

## 1.2 Classe: Bebida

- **Descrição:** Classe base para bebidas, tendo atribuindo comuns que serão usados por suas subclasses, incluindo atributos básicos como nome, preço, estoque, ML e se a bebida é alcoólica.

## 1.3 Classe: BebidaComAlcool

- **Descrição:** Extensão da classe Bebida, representa bebidas alcoólicas e possui métodos específicos para verificar a idade do consumidor.

## 1.4 Classe: BebidaSemAlcool

- **Descrição:** Extensão da classe Bebida, representa bebidas não alcoólicas, permitindo uma diferenciação nas operações.

## 1.5 Classe: Usuario:

- **Descrição:** Representa um usuário do sistema, contendo informações como nome, e-mail, CPF e data de nascimento. Inclui métodos para validar dados do usuário.

## 2. Classes de Controle:

### 2.1 Classe: BancoDeDados

- **Descrição:** Responsável pela conexão e interação com o banco de dados. Inclui métodos para operações CRUD (Create, Read, Update, Delete) sobre os dados.

### 2.2 Classe: UsuarioController

- **Descrição:** Gerencia a lógica de negócios relacionada aos usuários, como registro, login e remoção de administradores. Utiliza a classe BancoDeDados para persistir as informações.

### 2.3 Classe: AlimentoController

- **Descrição:** Gerencia operações relacionadas a alimentos permitindo tarefas, como adicionar, editar e remover alimentos do estoque. Utiliza a classe BancoDeDados para persistir as informações.

### 2.4 Classe: BebidaController

- **Descrição:** Gerencia operações relacionadas as duas bebidas permitindo tarefas, como adicionar, editar e remover alimentos do estoque. Utiliza a classe BancoDeDados para persistir as informações.

## 3. Interfaces de Usuário (Views):

### 3.1 Classe: Cadastro

- **Descrição:** Interface de usuário Swing para registro de novos usuários, permitindo a entrada de dados e validação.

### 3.2 Classe: CadastroAdmin

- **Descrição:** Interface para registrar administradores, com campos adicionais para definir privilégios.

### 3.3 Classe: Login

- **Descrição:** Interface de login que permite aos usuários autenticar-se no sistema, com validações adequadas.

### 3.4 Classe: dashAdmin

- **Descrição:** Painel administrativo que permite gerenciar usuários administradores e acessar diferentes seções do sistema.

### 3.5 Classe: dashCliente

- **Descrição:** Painel do cliente que permite gerenciar uma lista de compras e acessar seções de alimentos e bebidas.

### 3.6 Classe: AlimentoView

- **Descrição:** Interface para gerenciar alimentos, permitindo adicionar, editar e remover alimentos do estoque, além de validar entradas.

## 4. Interfaces:

### 4.1 Interface: ComAlcool

- **Descrição:** Interface que define métodos para bebidas alcoólicas.

### 4.2 Interface: ComPeso

- **Descrição:** Interface que define métodos para produtos que têm peso.

### 4.3 Interface: ComValor

- **Descrição:** Interface que define métodos para produtos que têm um valor associado.

### 4.4 Interface: ComVolume

- **Descrição:** Interface que define métodos para produtos que têm volume, aplicada a bebidas.

## 5. Exceções Personalizadas

### 5.1 Classe: CampoVazio

- **Descrição:** Exceção lançada quando campos obrigatórios estão vazios.

## 5.2 Classe: CpfEmUso

- **Descrição:** Exceção lançada quando um CPF já está cadastrado.

## 5.3 Classe: CpfInvalido

- **Descrição:** Exceção lançada quando um CPF não tem formato válido.

## 5.4 Classe: DataInvalida

- **Descrição:** Exceção lançada quando a data de nascimento é inválida.

## 5.5 Classe: EmailInvalido

- **Descrição:** Exceção lançada quando um email não está em um formato correto.

## 5.6 Classe: EmailJaCadastrado

- **Descrição:** Exceção lançada quando um email já está registrado no sistema.

## 5.7 Classe: ErroLogin

- **Descrição:** Exceção lançada quando há falha no login do usuário.

## 5.8 Classe: ErroValorNumerico

- **Descrição:** Exceção lançada quando um valor numérico é inserido de forma incorreta.

## 5.9 Classe: LoginInvalido

- **Descrição:** Exceção lançada quando as credenciais de login são inválidas.

# 4 - Instrução Para Compilação e Execução do Código

## Pré-requisitos:

1. Ter o Maven e o JDK instalados e configurados.
2. Descompactar o arquivo.

## Instruções:

1. Após descompactar, abra o **Prompt de Comando (CMD)** e navegue até a pasta onde você descompactou o arquivo.
2. Digite o comando **mvn clean compile** e pressione **Enter**.
3. Em seguida, digite o comando **mvn install** e pressione **Enter**.

Após executar esses comandos, uma pasta chamada **target** será criada no diretório do projeto. Essa pasta conterá o arquivo executável, que geralmente terá o nome do projeto e a extensão **SNAPSHOT-jar-with-dependencies.jar**.

**Para executar o programa:**

1. Va na pasta do projeto que você deseja executar;
2. abra o CMD ou terminal do Linux;
3. digite **mvn clean compile** no seu CMD para remover arquivos antigos;
4. digite **mvn install** após executar o passo anterior para instalar o projeto;
5. Execute o **java -jar target/SistemaMercado-1.0-SNAPSHOT-jar-with-dependencies.jar**

## **5 - Usuários de Acesso:**

**Administrador Chefe:**

- **Email:** admin@adminC.com
- **Senha:** admin123

**Administrador comum:**

- **Email:** admin@admin.com
- **Senha:** admin123

**Usuário (Menor de idade):**

- **Email:** cliente@cliente.com
- **Senha:** cliente123

**Usuário (Maior de idade):**

- **Email:** cliente@clienteM.com
- **Senha:** cliente123

## **6 - Interface Gráfica:**



# Click&Go

**"Click & Go: A Revolução da Compra Sem Fronteiras - Rápido, Fácil e Inteligente."**

**Bem-vindo ao Futuro das Compras com o Click & Go! Em nossa loja, a tecnologia de ponta se une à conveniência, oferecendo uma experiência de compra sem fricções. Sem caixas, sem filas— basta pegar o que precisa e sair. Nosso sistema inteligente usa sensores e inteligência artificial para identificar e cobrar seus itens automaticamente. Com o Click & Go, sua experiência de compra é rápida, segura e totalmente sem esforço. Venha descobrir a forma mais moderna e eficiente de fazer suas compras!**


Email:

Senha:

Entrar

Criar uma Conta

tela de login, do sistema Click&Go



# CLICK & GO

Nome:

Email:

Data de Nascimento:

setembro

▼

2018

▲

▼

	dom.	seg.	ter.	qua.	qui.	sex.	sáb.
35							1
36	2	3	4	5	6	7	8
37	9	10	11	12	13	14	15
38	16	17	18	19	20	21	22
39	23	24	25	26	27	28	29
40	30						

CPF

Senha:


**Criar Conta**

tela de cadastro do Cliente





dashboard do Administrador, mostrando todas as funcionalidades disponíveis



—

□

×

# CLICK & GO

Nome:

Email:

Data de Nascimento:

setembro

▼

2009

▲

▼

	dom.	seg.	ter.	qua.	qui.	sex.	sáb.
36			1	2	3	4	5
37	6	7	8	9	10	11	12
38	13	14	15	16	17	18	19
39	20	21	22	23	24	25	26
40	27	28	29	30			


CPF

Senha:

☐ Definir como ADM chefe

Criar Conta

Tela de Cadastro do Administrador

Voltar Logout

Arroz Branco 5 Kg

Feijao Preto 1 Kg

Açucar Cristal 1 Kg

Sal Refinado 1 Kg

Macarrao Espaguete 500g

Oleo de Soja 900ml

Cafe Torrado e Moido 500g

Molho de Tomate 340g

Lentilha 1 Kg

Farinha de Trigo 1 Kg

Nome:

Quantidade (estoque):

Peso (gramas):


Preco:

Adicionar

Editar

Remover

### Área de Controle de Alimentos

Voltar Logout

Bebidas Sem Alcool:

Refrigerante Coca-Cola 2L

Suco de Laranja Del Vale 1L

Água Mineral Crystal 1.5L

Chá Verde Lipton 1L

Refrigerante Guaraná Antártica 2L

Suco de Uva Concord 1L

Limonada Natural 1L

Refrigerante Sprite 2L

Água Tônica 1L

Suco de Abacaxi 1L

Bebidas ComAlcool:

Cerveja Brahma 350ml

Vinho Tinto Miolo 750ml

Vodka Absolut 1L

Whisky Johnnie Walker Red Label 750ml

Cerveja Skol 350ml

Cachaça 51 1L

Licor de Amaretto 700ml

Vinho Branco Salton 750ml

Cerveja Heineken 330ml

Espumante Chandon 750ml

Nome:

Estoque:

Quantidade (ML):

Preco:

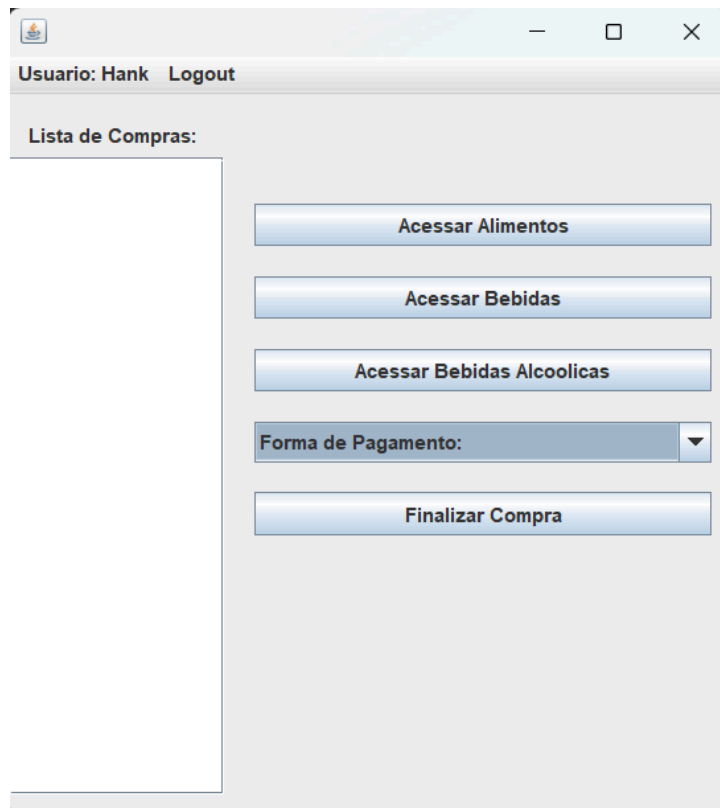
☐ Bebida Alcoolica

Adicionar

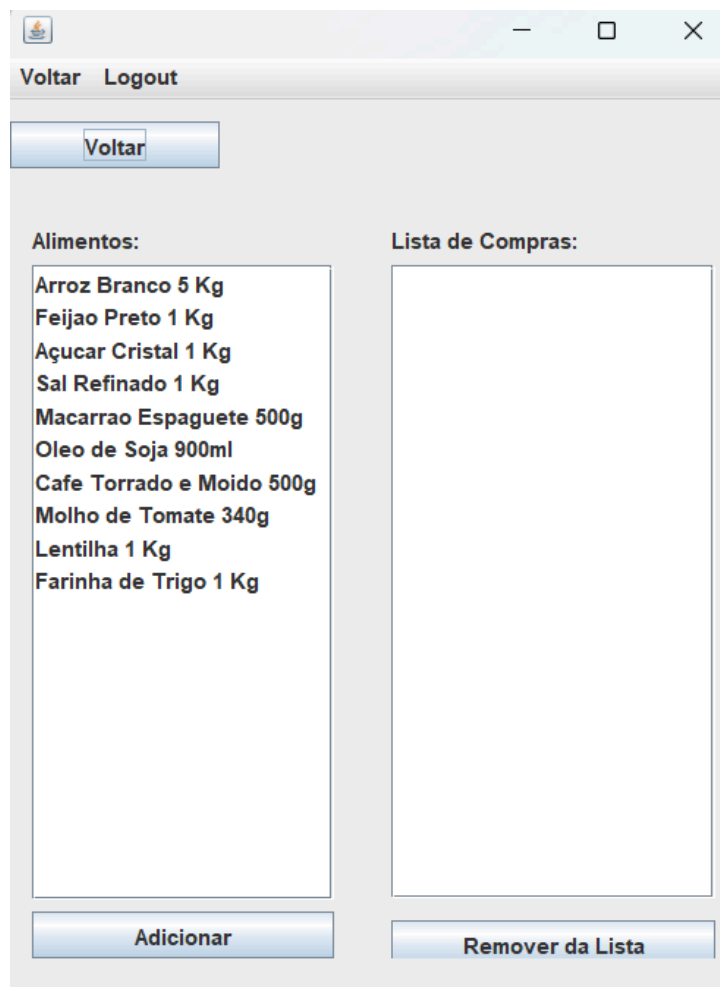
Editar

Remover

### Área de Controle de Bebidas



Dashboard do Cliente



Área de compra de Alimentos

Voltar

Logout

Voltar

Bebidas:

Refrigerante Coca-Cola 2L

Suco de Laranja Del Vale 1L

Água Mineral Crystal 1.5L

Chá Verde Lipton 1L

Refrigerante Guaraná Antárt

Suco de Uva Concord 1L

Limonada Natural 1L

Refrigerante Sprite 2L

Água Tônica 1L

Suco de Abacaxi 1L

Lista de Compras:

Adicionar

Remover da Lista

Voltar

Logout

Voltar

Bebidas Alcoolicas:

Cerveja Brahma 350ml

Vinho Tinto Miolo 750ml

Vodka Absolut 1L

Whisky Johnnie Walker Red

Cerveja Skol 350ml

Cachaça 51 1L

Licor de Amaretto 700ml

Vinho Branco Salton 750ml

Cerveja Heineken 330ml

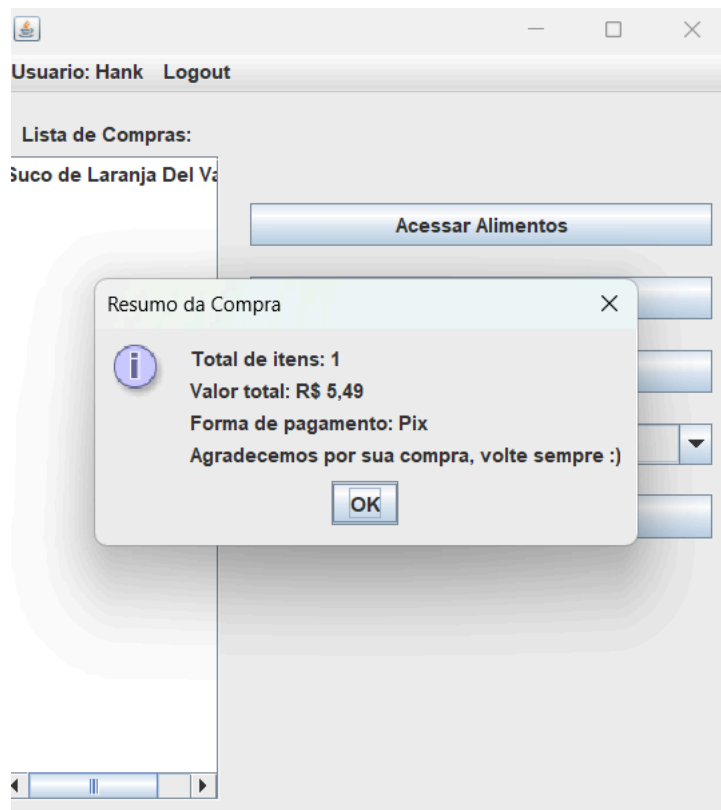
Espumante Chandon 750ml

Lista de Compras:

Adicionar

Remover da Lista

Área de compras de Bebidas e de Bebidas Alcoólicas



Finalização de compras

## Observações:

- Quando o admin for selecionar um produto é preciso dar click duplo para poder editar/deletar o produto por questões de segurança;
- Ao editar uma bebida não é possível mudar o seu tipo de alcoólica para não alcoólica;
- É sempre preciso ser um administrador chefe para poder deletar um admin;
- o Cadastro de usuário é liberado para crianças a partir de 6 anos de idade, mas a compra de bebidas alcoólicas é estritamente proibida para menores de idade;
- Compras efetuadas via credito sofrem um acréscimo de 2% devido as taxas da maquina;