

**UNIVERSIDADE ESTADUAL PAULISTA "JÚLIO DE MESQUITA FILHO"**

FACULDADE DE CIÊNCIAS - CAMPUS BAURU

DEPARTAMENTO DE COMPUTAÇÃO

BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

GABRIEL ZANFORLIN PARIZOTTO

JOÃO HENRIQUE MOURO SUAIKEN

LUCA MELO MUNEKATA

LUCAS YUKI NISHIMOTO

RAFAEL PICCOLOMINI DE LIMA

**PROJETO DE UM E-COMMERCE DE ROUPAS**

BAURU

Julho/2023

GABRIEL ZANFORLIN PARIZOTTO  
JOÃO HENRIQUE MOURO SUAIDEN  
LUCA MELO MUNEKATA  
LUCAS YUKI NISHIMOTO  
RAFAEL PICCOLOMINI DE LIMA

## **PROJETO DE UM E-COMMERCE DE ROUPAS**

Trabalho de Conclusão da disciplina de Engenharia de Software II do Curso de Ciência da Computação da Universidade Estadual Paulista “Júlio de Mesquita Filho”, Faculdade de Ciências, Campus Bauru.  
Orientador: Prof. Douglas Rodrigues

# Lista de figuras

Figura 1 – Esquema Arquitetura Microserviços . . . . .	8
Figura 2 – Diagrama de Classes . . . . .	10
Figura 3 – Legenda do Diagrama de Classes . . . . .	11
Figura 4 – Diagrama de Componentes . . . . .	12
Figura 5 – Modelo Entidade Relacionamento . . . . .	14
Figura 6 – Modelo Relacional . . . . .	15
Figura 7 – Interface Produtos . . . . .	17
Figura 8 – Interface Produto . . . . .	18
Figura 9 – Interface Login . . . . .	19
Figura 10 – Interface Carrinho . . . . .	20
Figura 11 – Interface Pagamento . . . . .	21
Figura 12 – Interface Perfil . . . . .	22

# Lista de tabelas

Tabela 1 – Tabela cliente . . . . .	15
Tabela 2 – Tabela produto . . . . .	15
Tabela 3 – Tabela avaliacao . . . . .	16
Tabela 4 – Tabela item carrinho . . . . .	16
Tabela 5 – Tabela seleciona . . . . .	16
Tabela 6 – Tabela carrinho . . . . .	16
Tabela 7 – Tabela pagamento . . . . .	16

# Lista de abreviaturas e siglas

SGBD	Sistema de Gerenciamento de Banco de Dados
MER	Modelo Entidade Relacionamento
MR	Modelo Relacional

# Sumário

1	INTRODUÇÃO . . . . .	6
2	ANÁLISE DE REQUISITOS . . . . .	7
3	ARQUITETURA . . . . .	8
4	DIAGRAMA DE CLASSES . . . . .	10
5	DIAGRAMA DE COMPONENTES . . . . .	12
6	BANCO DE DADOS . . . . .	14
6.1	Modelo Entidade Relacionamento . . . . .	14
6.2	Modelo Relacional . . . . .	15
6.3	Tabelas . . . . .	15
7	INTERFACES . . . . .	17
7.1	Interface Produtos . . . . .	17
7.2	Interface Página de Produto . . . . .	18
7.3	Interface Login . . . . .	19
7.4	Interface Carrinho . . . . .	20
7.5	Interface Pagamento . . . . .	21
7.6	Interface Perfil . . . . .	22
8	TESTES . . . . .	23
8.1	Teste de unidade . . . . .	23
8.2	Teste de integração . . . . .	23
8.3	Teste de validação . . . . .	23
8.4	Teste de sistema . . . . .	24
8.5	Teste de segurança . . . . .	24

# 1 Introdução

O Aplicativo de e-Commerce de roupas é um sistema desenvolvido para oferecer uma plataforma de compras online exclusiva da nossa loja. Através do aplicativo, os clientes podem explorar e adquirir os produtos de moda que oferecemos de forma conveniente e segura. O objetivo principal do aplicativo é proporcionar uma experiência de compra agradável e eficiente para os usuários.

No aplicativo, os clientes têm acesso a um catálogo online abrangente, onde podem navegar e pesquisar por diferentes categorias de roupas e acessórios. Cada produto é apresentado com informações detalhadas, imagens e preços atrativos. Os clientes podem adicionar os produtos desejados ao seu carrinho de compras enquanto exploram o catálogo. Eles podem revisar os itens no carrinho, atualizar as quantidades e remover produtos antes de prosseguir para o checkout.

Para facilitar o processo de pagamento, o aplicativo suporta diversos métodos de pagamento seguros, como cartões de crédito e carteiras digitais. Os clientes podem fornecer as informações necessárias de pagamento de forma segura e concluir a transação com confiança. Além disso, o aplicativo oferece um perfil personalizado para cada cliente, permitindo que eles acompanhem o histórico de pedidos, o status de entrega e realizem solicitações de trocas ou devoluções, caso necessário.

Valorizamos a opinião dos nossos clientes e incentivamos o compartilhamento de experiências. No aplicativo, os clientes podem deixar avaliações e comentários sobre os produtos que adquiriram, permitindo que outros usuários tomem decisões de compra informadas. Essas avaliações fornecem feedback valioso para nós e ajudam a melhorar a qualidade dos nossos produtos e serviços.

Em resumo, o Aplicativo de e-Commerce de roupas é uma plataforma exclusiva da nossa loja para oferecer uma experiência de compra agradável e prática aos clientes. Através do aplicativo, os clientes podem acessar o catálogo de produtos, adicionar itens ao carrinho de compras, efetuar pagamentos seguros e deixar avaliações dos produtos adquiridos. Estamos comprometidos em fornecer um serviço de alta qualidade aos nossos clientes, tornando o processo de compra de roupas online fácil e satisfatório.

## 2 Análise de Requisitos

Durante a análise dos requisitos para o Aplicativo de e-Commerce de Roupas, foram identificadas diversas funcionalidades essenciais para proporcionar uma experiência de compra satisfatória aos clientes. Um dos requisitos principais é o cadastro de clientes, permitindo que eles criem uma conta no aplicativo fornecendo informações pessoais, como nome, endereço de e-mail e senha. Além disso, o aplicativo deve possibilitar o login e logout para que os clientes possam acessar suas contas de maneira segura.

Outro requisito fundamental é o catálogo de produtos, que deve apresentar uma ampla variedade de roupas e acessórios disponíveis para compra. Cada produto deve ser detalhado com informações como nome, descrição, preço, tamanhos disponíveis e imagens ilustrativas. Os clientes devem poder navegar pelo catálogo, utilizar filtros por categoria, realizar buscas por palavras-chave e acessar páginas individuais de cada produto para obter mais detalhes.

O carrinho de compras é uma funcionalidade essencial para permitir que os clientes selecionem e organizem os produtos desejados. Eles devem ser capazes de adicionar itens ao carrinho, visualizar um resumo dos produtos selecionados e ter a opção de atualizar a quantidade, remover itens específicos ou limpar completamente o carrinho. É importante que o carrinho de compras mantenha os produtos selecionados mesmo que o cliente saia e retorne posteriormente ao aplicativo.

O processamento de pagamentos é um requisito crítico para garantir transações financeiras seguras. O aplicativo deve oferecer opções confiáveis de pagamento, como cartões de crédito e carteiras digitais, e garantir que as informações de pagamento sejam protegidas e transmitidas de forma criptografada.

Cada cliente deve ter um perfil pessoal onde possa visualizar e atualizar suas informações, como endereço de entrega, dados de pagamento e histórico de pedidos. Além disso, o perfil do cliente deve permitir que ele acompanhe o status dos pedidos em andamento, rastreie entregas e solicite trocas ou devoluções, se necessário.

Uma funcionalidade importante é a possibilidade de os clientes deixarem avaliações e comentários sobre os produtos adquiridos. Essas avaliações fornecem feedback valioso para outros clientes e podem influenciar suas decisões de compra. O aplicativo deve exibir as avaliações de forma clara, incluindo a classificação por estrelas e os comentários dos clientes.

Por fim, é interessante explorar a capacidade do aplicativo de oferecer recomendações personalizadas de produtos com base no histórico de compras e nas preferências dos clientes. Essas recomendações podem melhorar a experiência de compra, mostrando itens relevantes e adequados aos gostos individuais de cada cliente.

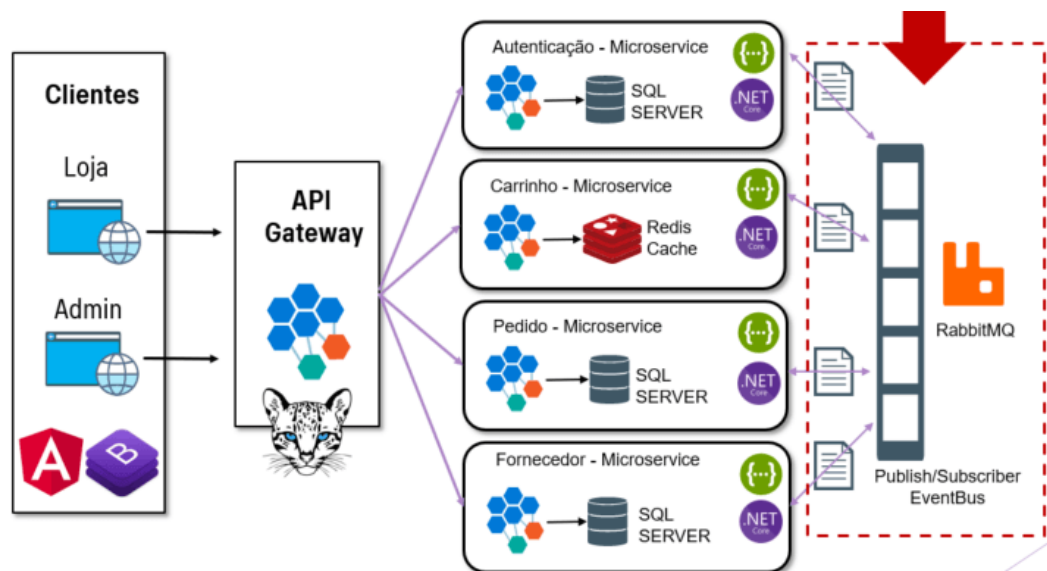


### 3 Arquitetura

Nesta seção, mostraremos o projeto arquitetural, lembrando que o site é um e-commerce, e buscando a opção que melhor pode atender as necessidades, a arquitetura escolhida foi a de Microserviços.

Primeiramente é válido ressaltar a importância de um projeto arquitetural na engenharia de software já que o projeto de arquitetura é crucial para garantir a escalabilidade, a manutenibilidade e o sucesso geral do desenvolvimento de software.

Figura 1 – Esquema Arquitetura Microserviços



Fonte: <https://marcdias.com.br/microservicos-conceitos-e-tecnologias-envolvidas-neste-estilo-arquitetural-de-sistemas-parte-2/>

Quando falamos em microserviços, estamos nos referindo a uma funcionalidade que pode ser dividida em partes menores. Desse modo, essas pequenas partes se comunicam por meio de uma interface bem definida, por exemplo APIs.

Como são executados de forma independente, cada serviço pode ser atualizado e implantado para atender às demandas de uma aplicação. A utilização de microserviços pode trazer muita produtividade, pois você pode desenvolver vários microserviços ao mesmo tempo, com diversos desenvolvedores trabalhando de forma simultânea na mesma aplicação com abordagens diferentes (como tecnologias envolvidas).

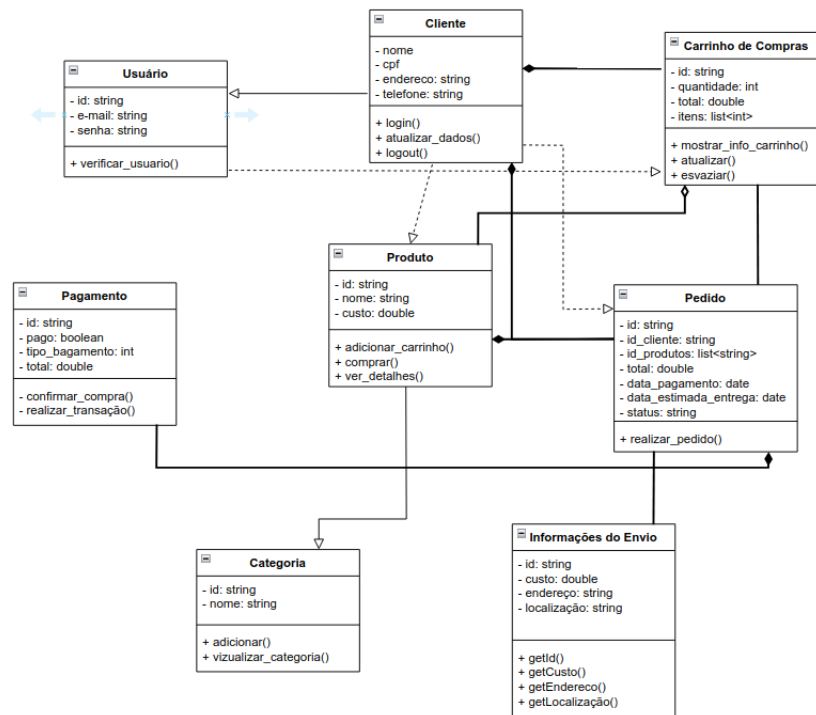
Vantagens:

- agilidade
- baixo acoplamento

- escalabilidade flexível
- flexibilidade para implantação de tecnologias heterogêneas

## 4 Diagrama de Classes

Figura 2 – Diagrama de Classes

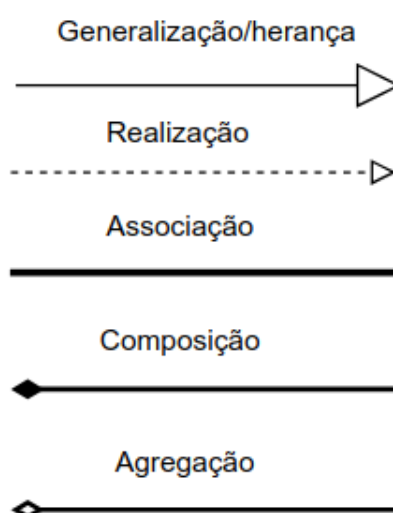


Fonte: Elaborado pelo autor.

- I. **Usuário**: esta classe representará o potencial cliente da loja, que pode navegar entre os produtos e seleciona-los e adiciona-los ao carrinho de compras.
- II. **Cliente**: esta classe contém informações mais pessoais do usuário e consegue efetuar login e logout, além de comprar produtos individuais ou do carrinho de compras.
- III. **Carrinho de compras**: esta classe representa os itens de desejo dos usuários, contendo informações de quantidade, custo e itens dentro do carrinho.
- IV. **Produto**: esta classe representa um produto da loja e possui atributos como preço e nome.
- V. **Categoria**: esta classe representa as diferentes categorias que podem ter entre os itens da loja.
- VI. **Pagamento**: esta classe realiza o pagamento de um ou mais produtos e possui atributos como tipo de pagamento e pago, que verifica se os produtos foram pagos.

- VII. Pedido: através da classe Pedido é possível fazer o pedido de um ou mais produtos.
- VIII. Informações de envio: esta classe possui informações sobre o envio de um pedido, como custo, endereço de destino e localização atual dos itens.

Figura 3 – Legenda do Diagrama de Classes



Fonte: chiu.

Herança é um conceito na UML em que uma classe pode herdar características de uma classe pai, também conhecida como superclasse. A classe filha, ou subclasse, herda os atributos e métodos da superclasse, permitindo a reutilização de código e estabelecendo uma relação de especialização.

Realização é um conceito na UML que descreve a relação entre uma classe/interface e uma classe/interface mais genérica. A classe/interface mais específica implementa total ou parcialmente as funcionalidades definidas na classe/interface mais genérica.

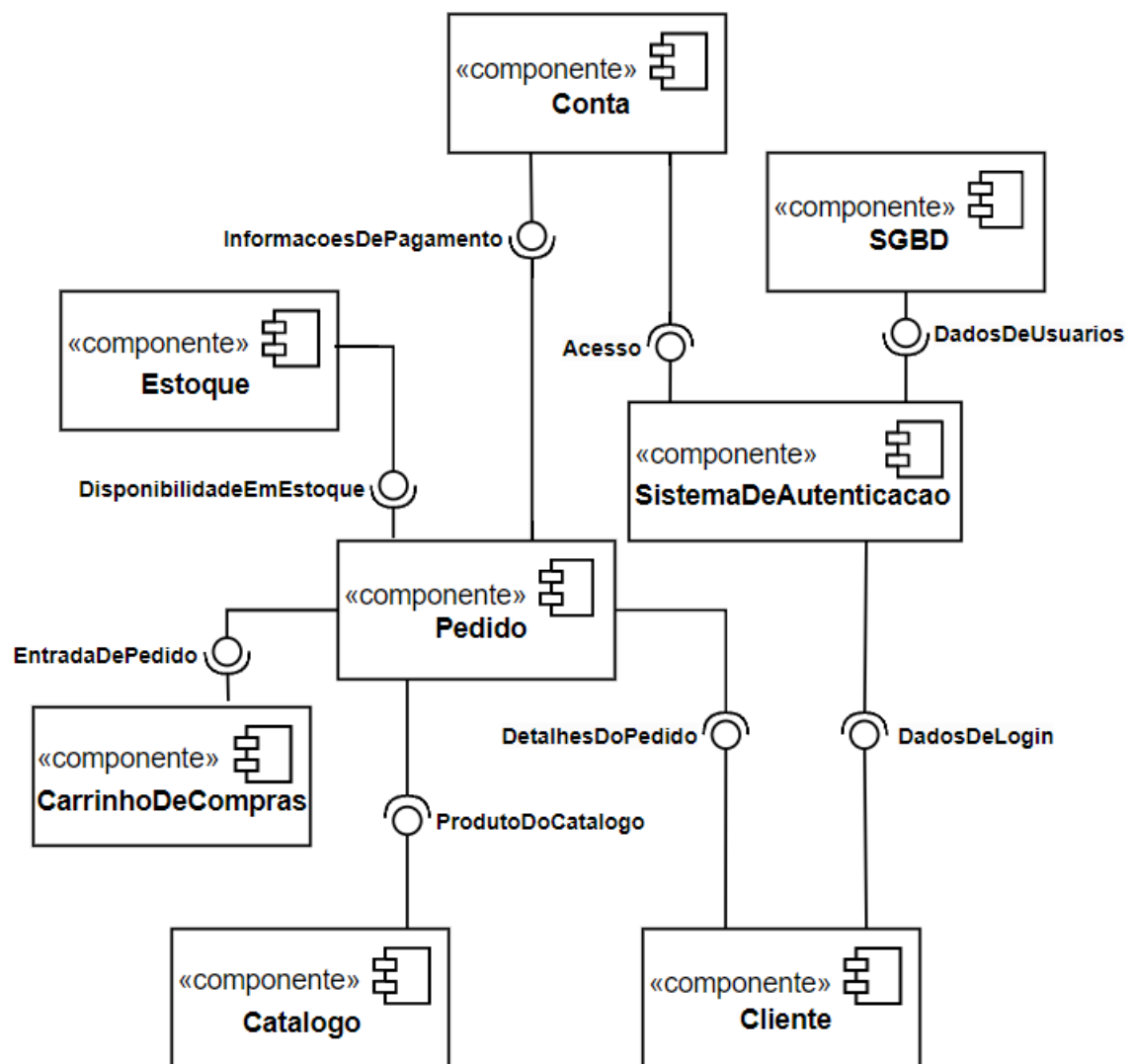
A associação é um relacionamento entre duas classes na UML, indicando que objetos de uma classe estão conectados a objetos de outra classe. Ela pode ser unidirecional ou bidirecional, e pode ter multiplicidades que indicam a quantidade de objetos que podem se relacionar.

A composição é um tipo especial de associação em que uma classe é composta por outras classes. Existe uma relação de todo-parte, em que a parte não pode existir independentemente do todo. Quando o todo é destruído, as partes também são.

A agregação é outro tipo de associação que representa um relacionamento de parte-todo, mas de forma mais fraca do que a composição. Nesse caso, as partes podem existir independentemente do todo e podem ser compartilhadas por outros objetos.

## 5 Diagrama de Componentes

Figura 4 – Diagrama de Componentes



Fonte: Elaborado pelo autor.

- **SistemaDeAutenticacao:** Este componente lida com o processo de autenticação dos usuários do e-commerce. Ele verifica os dados fornecidos pelo cliente durante o login (em associação com o SGBD) e é usado para autorizar as requisições feitas pelo cliente.
- **SGBD:** Este componente é responsável por armazenar e gerenciar os dados do e-commerce, como informações de contas de usuários. O SGBD fornece recursos para inserir, consultar, atualizar e excluir dados.
- **Cliente:** Este componente representa o usuário final do e-commerce, que acessa o sistema para visualizar e comprar produtos.
- **Conta:** Este componente é responsável por gerenciar as informações relacionadas à conta do usuário. Ele permite que o cliente atualize seus dados pessoais, como endereço de entrega e informações de pagamento. A conta do usuário também pode fornecer recursos adicionais, como a criação e o gerenciamento de listas de desejos, avaliação de produtos e outras personalizações. O componente de conta do usuário se comunica com o sistema de autenticação e os pedidos.
- **Pedido:** Este componente é responsável pelo gerenciamento dos pedidos feitos pelos clientes. Ele permite adicionar itens ao carrinho de compras, finalize a compra, visualize o histórico de pedidos, etc. O componente se comunica com outros, como o catálogo de produtos e o serviço de pagamento.
- **Estoque:** Este componente é responsável por gerenciar o estoque de produtos disponíveis para venda. Ele verifica a quantidade de cada produto em estoque e atualiza o estoque conforme as vendas são realizadas. O componente estoque pode incluir funcionalidades como a notificação de estoque baixo e se comunica com o catálogo de produtos e os pedidos.
- **Catalogo:** Este componente é responsável por armazenar e gerenciar informações sobre os produtos disponíveis no e-commerce. Ele fornece métodos de consulta para recuperar informações detalhadas sobre os produtos, como descrições, preços, tamanhos, etc.
- **CarrinhoDeCompras:** Este componente é responsável por gerenciar os itens selecionados pelo cliente para compra. Ele permite que o cliente adicione e remova produtos do carrinho, altere a quantidade de peças e calcule o subtotal da compra. O carrinho de compras também pode oferecer funcionalidades adicionais, como aplicação de cupons de desconto, cálculo de frete e estimativa do total da compra. O componente de carrinho de compras interage com o catálogo de produtos e os pedidos.

## 6 Banco de Dados

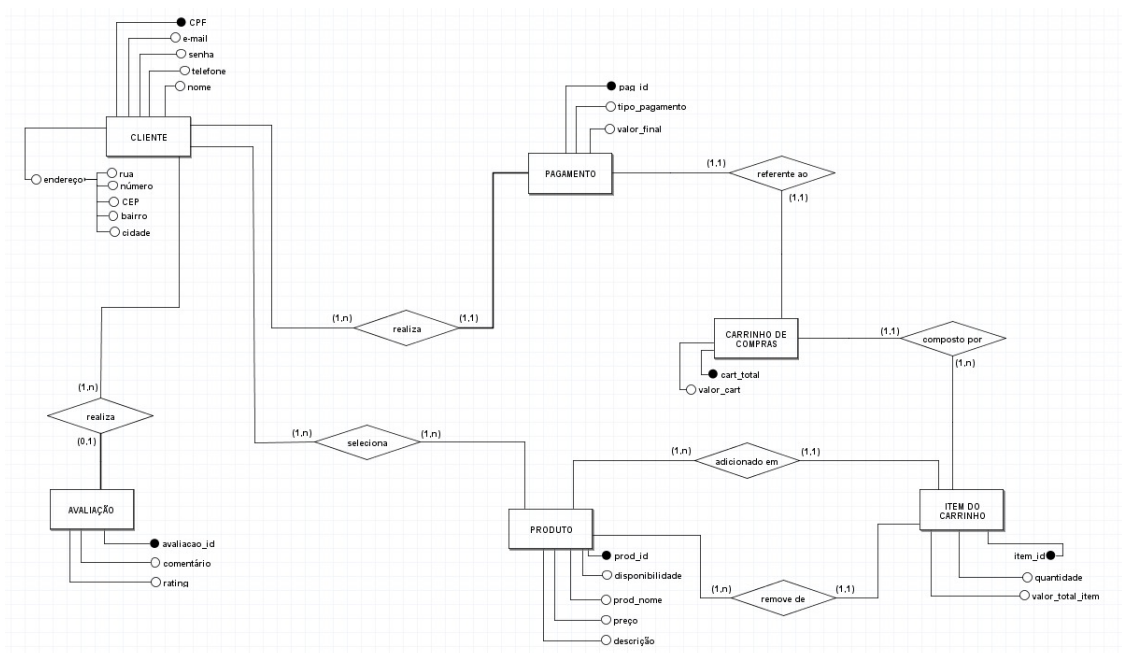
Neste capítulo, será apresentada a estrutura do banco de dados do aplicativo de e-commerce, que serve como base para o armazenamento e gerenciamento das informações relacionadas aos produtos, clientes, pedidos e demais elementos do sistema. Serão explorados os modelos conceituais MER (Modelo Entidade-Relacionamento) e MR (Modelo Relacional), além de fornecer uma visão geral das tabelas que compõem o banco de dados.

### 6.1 Modelo Entidade Relacionamento

O Modelo Entidade-Relacionamento (MER) é uma técnica utilizada para representar a estrutura lógica do banco de dados em termos de entidades, atributos e relacionamentos entre eles. O MER é um modelo conceitual que ajuda a compreender e planejar a organização dos dados antes de implementá-los no modelo relacional.

No MER do aplicativo de e-commerce, identificamos as principais entidades envolvidas, como Produto, Cliente, Pedido e Avaliação. Além disso, definimos os atributos de cada entidade, como nome, descrição, preço, e-mail, endereço, entre outros, que são necessários para armazenar as informações relevantes.

Figura 5 – Modelo Entidade Relacionamento

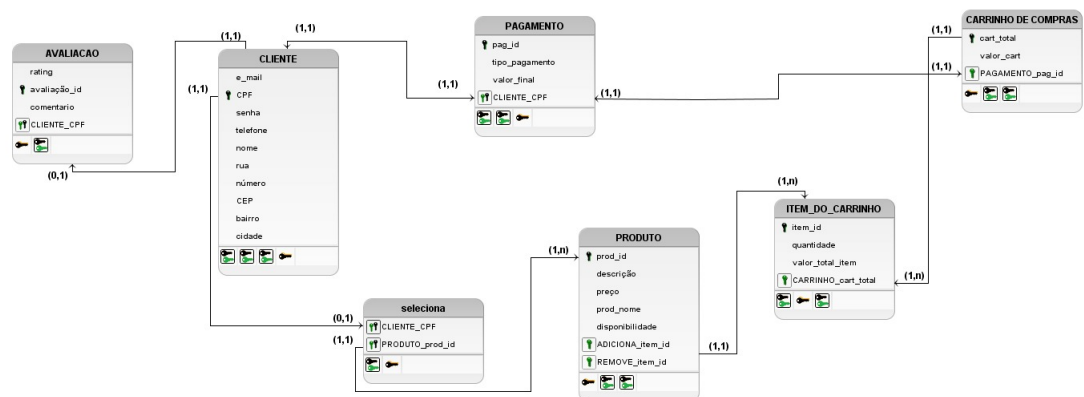


Fonte: Elaborado pelo autor.

## 6.2 Modelo Relacional

A partir do modelo conceitual MER, realizamos a etapa de relacionalização, que consiste em transformar as entidades, atributos e relacionamentos em tabelas, colunas e chaves em um banco de dados relacional. O Modelo Relacional(MR) é a representação concreta do banco de dados, onde as tabelas são criadas para armazenar os dados de forma organizada.

Figura 6 – Modelo Relacional



Fonte: Elaborado pelo autor.

## 6.3 Tabelas

Por fim, o banco de dados do aplicativo de e-commerce é composto por diversas tabelas, cada uma responsável por armazenar informações específicas. As tabelas formadas são: "produto" para armazenar os dados dos produtos, "cliente" para armazenar os dados dos clientes, "avaliacao" para armazenar as avaliações dos produtos feitas pelos clientes, "seleciona" para relacionar um cliente a um produto, "pagamento" para armazenar as informações do pagamento de um pagamento, "carrinho" para armazenar os itens no carrinho de um cliente e o "item carrinho" para armazenar um tipo específico do item que está no carrinho.

Tabela 1 – Tabela cliente

cpf	email	nome	senha	telefone	rua	numero	cep	bairro	cidade
-	-	-	-	-	-	-	-	-	-

A tabela cliente possui como chave primária o cpf e possui informações básicas do cliente, como nome, senha, email, cpf, entre outros.

Tabela 2 – Tabela produto

produto_id	descricao	preco	produto_nome
-	-	-	-



A tabela produto possui chave primária um id do produto e possui atributos como nome, preço e descrição.

Tabela 3 – Tabela avaliacao

<b>avaliacao_id</b>	rating	comentario	<b>cliente_cpf</b>
-	-	-	-

A tabela avaliação armazenará a avaliação de um cliente, por isso possui chave primária o id da avaliação e uma chave estrangeira para referenciar o cliente que fez a avaliação através do cpf.

Tabela 4 – Tabela item carrinho

<b>item_id</b>	quantidade	valor_total_item	<b>carrinho_id</b>
-	-	-	-

A tabela item carrinho possui um id como chave primária e armazena a informação de um determinado item no carrinho, por isso possui uma chave estrangeira "carrinho\_id" que referencia uma tabela carrinho.

Tabela 5 – Tabela seleciona

<b>cliente_cpf</b>	<b>produto_id</b>
-	-

Essa tabela tem a funcionalidade de indicar que um cliente escolheu um produto, por isso sua chave primária é composta por duas chaves estrangeiras: cliente\_cpf e produto\_id.

Tabela 6 – Tabela carrinho

<b>carrinho_id</b>	valor_carrinho	<b>pagamento_id</b>
-	-	-

Essa tabela indica o carrinho de compras total de um cliente, e possui um id como chave primária e uma chave estrangeira pagamento\_id para que seja possível realizar o pagamento do carrinho.

Tabela 7 – Tabela pagamento

<b>pagamento_id</b>	tipo_pagamento	valor_final	<b>cliente_cpf</b>
-	-	-	-

Por fim, a tabela pagamento serve para armazenar dados do pagamento de um pedido, sendo que a chave primária é um id do pagamento e a chave estrangeira cliente\_cpf referencia o cpf do cliente que está realizando a compra.

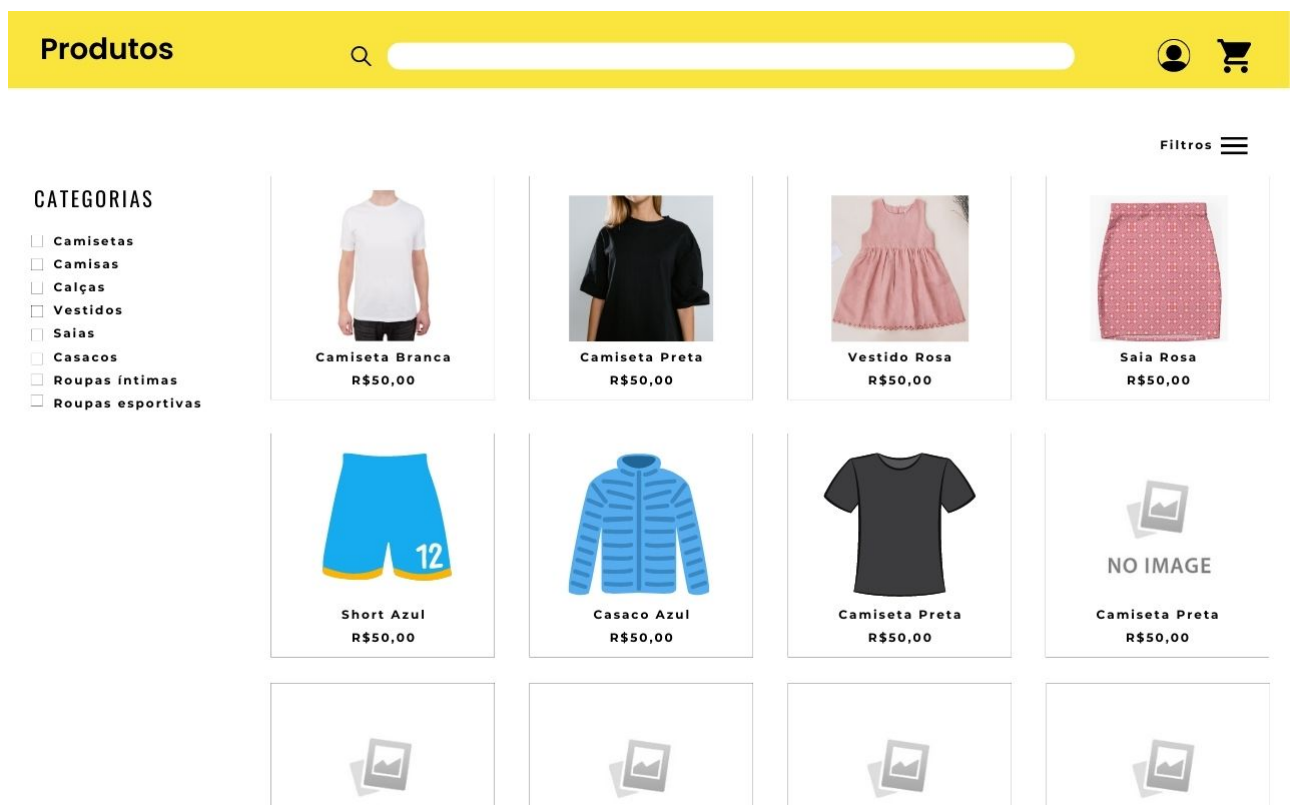
## 7 Interfaces

Os esboços das principais interfaces do projeto seguem abaixo. Elas visam atender as chamadas Regras de Ouro cunhadas por Theo Mandel.

### 7.1 Interface Produtos

Interface responsável por exibir todos os produtos da lista, permitir filtro por categoria e ordenação.

Figura 7 – Interface Produtos



Fonte: Elaborado pelo autor.

## 7.2 Interface Página de Produto

Esta interface tem como objetivo apresentar o produto escolhido pelo cliente, disponibilizando todas as características do produto e oferecendo a opção para compra ou adição no carrinho

Figura 8 – Interface Produto

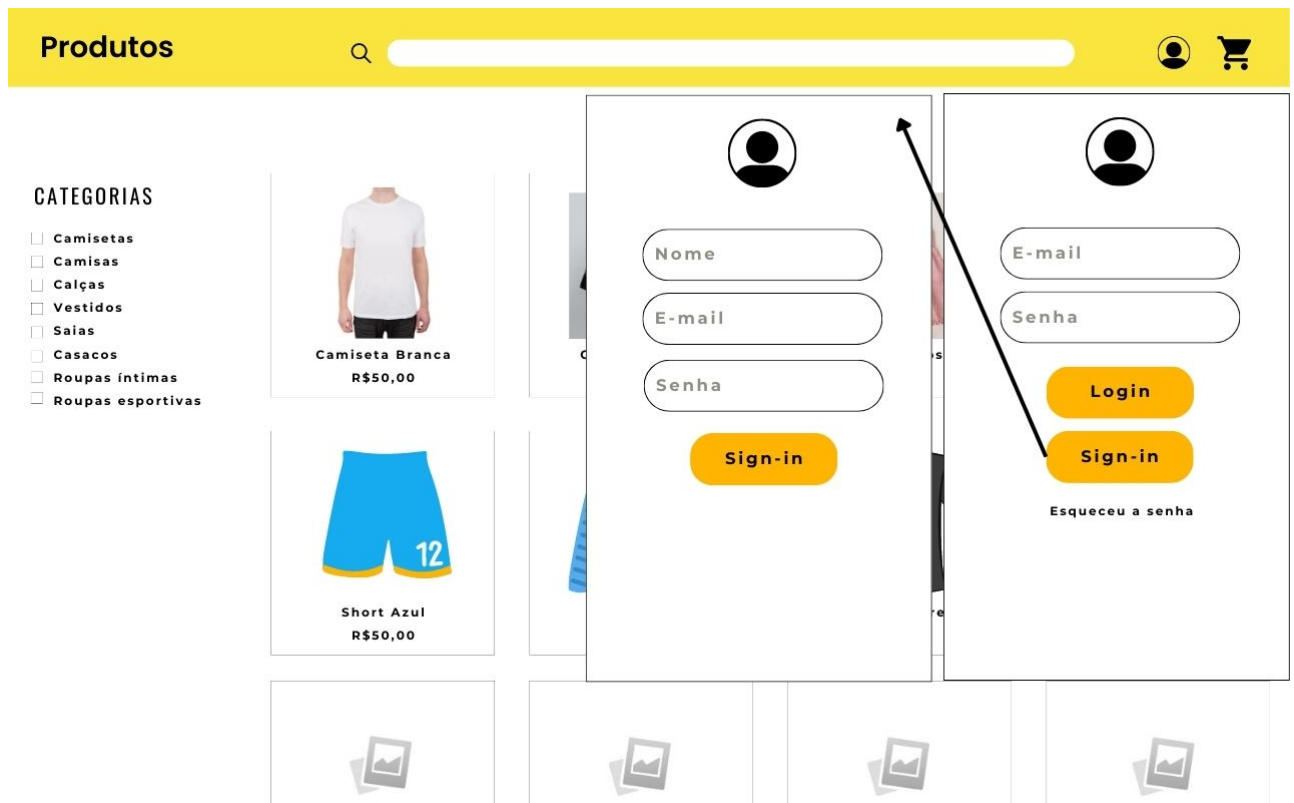


Fonte: Elaborado pelo autor.

### 7.3 Interface Login

Responsável por efetuar o login e cadastro do cliente esta interface utiliza de uma área flutuante para uma utilização fácil e rápida, bastando a apenas preencher os campos e clicar no botão a ser utilizado.

Figura 9 – Interface Login

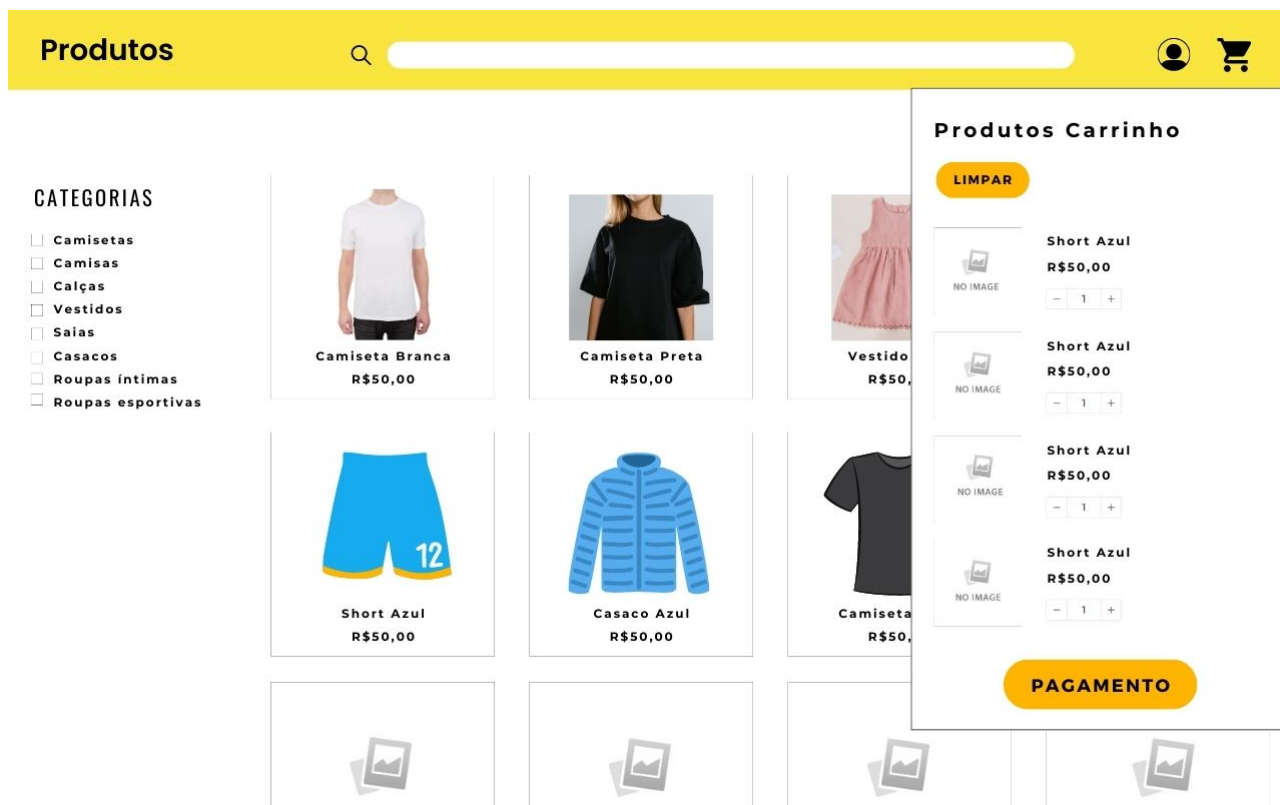


Fonte: Elaborado pelo autor.

## 7.4 Interface Carrinho

Também utilizando de uma área flutuantes, a interface do carrinho permite a visualização de todos os produtos adicionados, remoção e adição, uma função de limpar o carrinho por completo e um botão que leva o cliente para a tela de pagamento.

Figura 10 – Interface Carrinho




Fonte: Elaborado pelo autor.


## 7.5 Interface Pagamento

Sendo direcionado pelo carrinho ou página do produto, o cliente entrará na página de pagamento. Nesta página poderá conferir todos os produtos do pedido, verificar e alterar o endereço de entrega e definir qual método de pagamento será utilizado.


Figura 11 – Interface Pagamento

**Pagamento**

**Produtos**

  
NO IMAGE

**Short Azul**  
**R\$50,00**  
- 1 +

  
NO IMAGE

**Short Azul**  
**R\$50,00**  
- 1 +


**Total do Pedido: R\$100,00**


**Endereço de Entrega**


R. Manoel Pereira Rolla, 11-90 -  
Vila Nova Cidade Universitária, Bauru - SP, 17012-190  
Jonas Marcelo da Costa

**Alterar Endereço**

**Forma de pagamento**

  
Cartão de Credito  
☐

  
PIX  
☒

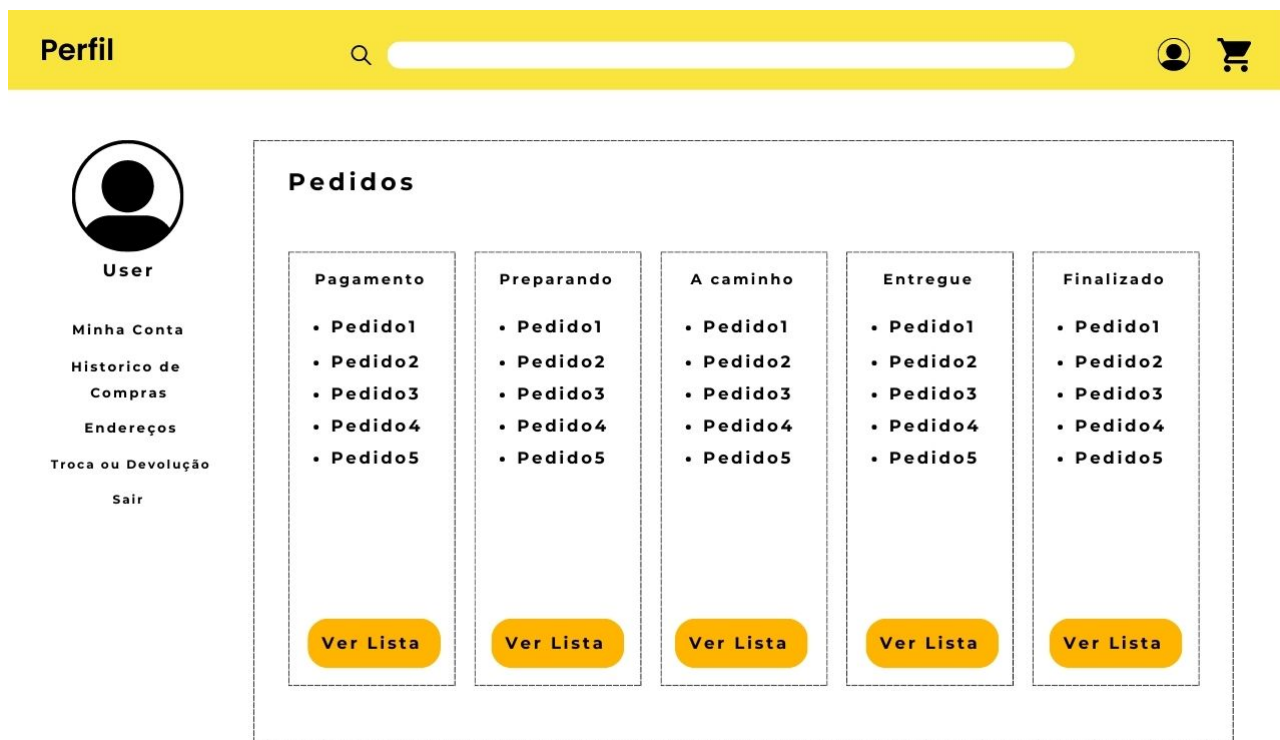
  
Boleto  
☐

Fonte: Elaborado pelo autor.

## 7.6 Interface Perfil

Uma vez que o cliente esteja logado na loja, sempre que clicar no botão de perfil será direcionado para esta interface a qual oferece todo o gerenciamento dos pedidos, alterações da conta, endereços, abrir chamados de troca ou devolução e sair do sistema.

Figura 12 – Interface Perfil



Fonte: Elaborado pelo autor.

## 8 Testes

A fase de testes corresponde a uma parte fundamental do desenvolvimento de software. É nela que os erros do sistema devem ser revelados e, deste modo, possam ser tratados. Entretanto não é algo simples, para que todos os erros sejam detectados é preciso se estabelecer uma estratégia eficaz e sistemática.

### 8.1 Teste de unidade

Corresponde ao teste de cada unidade ou componente individualmente. Nessa etapa é testado os códigos de cada componente e se este está funcionando corretamente. O primeiro componente a ser testado será o cadastro dos clientes. Testaremos então a funcionalidade do cadastro, o que acontece se tentar cadastrar dois clientes com os mesmos dados, a verificação da senha do usuário, o armazenamento das informações dos clientes, a alteração dos dados entre outros. Outros componentes também devem ser testados, como o catálogo de produtos, que deve exibir os produtos de forma organizada, nomeados e sem itens repetidos. O pagamento também corresponde a uma parte fundamental, já que o pagamento deve ser processado, um comprovante de ser gerado, assim como a ordem de venda. O carrinho deve ser capaz de armazenar, remover e adicionar itens, mostrar o valor do carrinho e mostrar o que há nele.

### 8.2 Teste de integração

Nesta parte os componentes devem estar interagindo entre si. Por isso, é preciso testar a interação do catálogo com o carrinho, se os itens estão sendo adicionados no carrinho, o carrinho também deve ser capaz de gerar um pedido que posteriormente irá para a etapa do pagamento. Tanto no pedido quanto no pagamento é preciso que as informações do cliente estejam registradas, informações essas que serão regatadas do cadastro do usuário. O cliente deve ser capaz de interagir com o catálogo de produtos, clicando neles, passando as fotos, vendo informações entre outros.

### 8.3 Teste de validação

Aqui o projeto é testado e seus requisitos já devem estar implementados e em funcionamento, deve ser possível se cadastrar, realizar compras e pagamentos, vasculhar pela loja, utilizar o sistema de busca, acompanhar o status do pedido e fornecer avaliações.



## 8.4 Teste de sistema

Neste ponto o teste não se concentra apenas no software em si, mas em todo o contexto em que ele está inserido. Deve ter em mente como as pessoas utilizarão esse software, necessidades que elas podem vir a ter, a interação do software com o estoque da loja, o funcionamento do banco de dados, etc. Essa deve ser uma fase que está em constante aprimoramento e não tem um fim definido.

## 8.5 Teste de segurança

Nesse passo é preciso verificar se o sistema está protegido contra ameaças de segurança, como ataques de injeção de SQL, cross-site scripting (XSS) e falsificação de solicitação entre sites (CSRF). É importante proteger os dados dos clientes assim como assegurar uma forma de pagamento confiável.