

CENTRO PAULA SOUZA  
FACULDADE DE TECNOLOGIA DE ITAPETININGA  
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE  
SISTEMAS

FELIPE SOARES CIRINEU  
GABRIEL FIDELIS PINTO  
GUSTAVO TSUTOMU KOIZUMI  
MARCELO AUGUSTO CACHETA

PROJETO ENG. SOFTWARE E PROG. WEB: E-COMMERCE PARA PEQUENOS  
EMPREendedores

**Itapetininga, SP**

**2º Semestre / 2025**

## **1. DESCRIÇÃO DO PROJETO**

O projeto tem como objetivo desenvolver um sistema simples de e-commerce voltado para pequenos comércios locais, permitindo que lojistas cadastrem produtos e clientes possam realizar compras online. O sistema busca modernizar o comércio tradicional, trazendo mais visibilidade e praticidade para vendedores e consumidores.

Além disso, o aplicativo em PHP foi programado para, ao finalizar o pedido, gerar automaticamente uma mensagem pronta com os detalhes da compra e enviá-la para o número de WhatsApp da empresa contratante, facilitando a comunicação e o atendimento dos pedidos.

## **2. USO JUSTIFICADO DE NOSQL**

O módulo de carrinho do sistema utiliza um banco de dados NoSQL de chave–valor local, por meio do LocalStorage do navegador. Diferente de um banco NoSQL de servidor (como MongoDB ou Firebase), o LocalStorage atua no cliente, armazenando os dados diretamente no navegador do usuário. Isso significa que o carrinho não é salvo no banco relacional (MySQL), mas em um armazenamento local e independente do servidor.

O uso do LocalStorage foi adotado para melhorar a performance e a experiência do usuário, permitindo que:

- Os produtos adicionados ao carrinho sejam salvos mesmo sem conexão com o servidor;
- O usuário possa fechar o navegador e retornar mais tarde sem perder os itens;
- O sistema funcione de forma rápida e responsiva, sem necessidade de chamadas ao banco de dados MySQL até a finalização do pedido.

## **X. TRATAMENTO DE EXCESSOES**

O tratamento principal é feito com blocos *try/catch* e validações condicionais, especialmente em api.php e nos models (como Order.php, Product.php, User.php). O sistema utiliza tratamento de exceções estruturado em PHP e JavaScript para prevenir erros, proteger dados e melhorar a experiência do usuário.

E o core/Errorhandler.php que é um componente de tratamento de exceções global, que garante a estabilidade, segurança e rastreabilidade de erros no sistema. O arquivo core/Errorhandler.php centraliza o tratamento global de erros e exceções no sistema.

Ou seja, mesmo que uma exceção não seja capturada em um *try/catch* local, esse script garante que o erro será registrado, tratado e comunicado de forma controlada ao usuário e ao desenvolvedor.

## X. USO DE DESIGN SYSTEM (EXTRA)

O projeto utiliza o Bootstrap Icons como parte do seu *design system*, oferecendo um conjunto padronizado, leve e escalável de ícones para a interface do sistema.

A biblioteca é incorporada ao projeto por meio de um link CDN (*Content Delivery Network*), conforme o exemplo abaixo:

```
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.11.3/font/bootstrap-icons.css">
```

O CDN é uma rede de distribuição de conteúdo que permite carregar arquivos externos diretamente da internet, sem a necessidade de armazená-los localmente no servidor do projeto. Esse método melhora o desempenho, a velocidade de carregamento e a eficiência do sistema, pois utiliza servidores otimizados e geograficamente distribuídos para entregar o conteúdo de forma mais rápida e confiável.

### 3. Arquiteturas

#### 3.1 Arquitetura – MVC

A arquitetura MVC divide o sistema em três camadas:

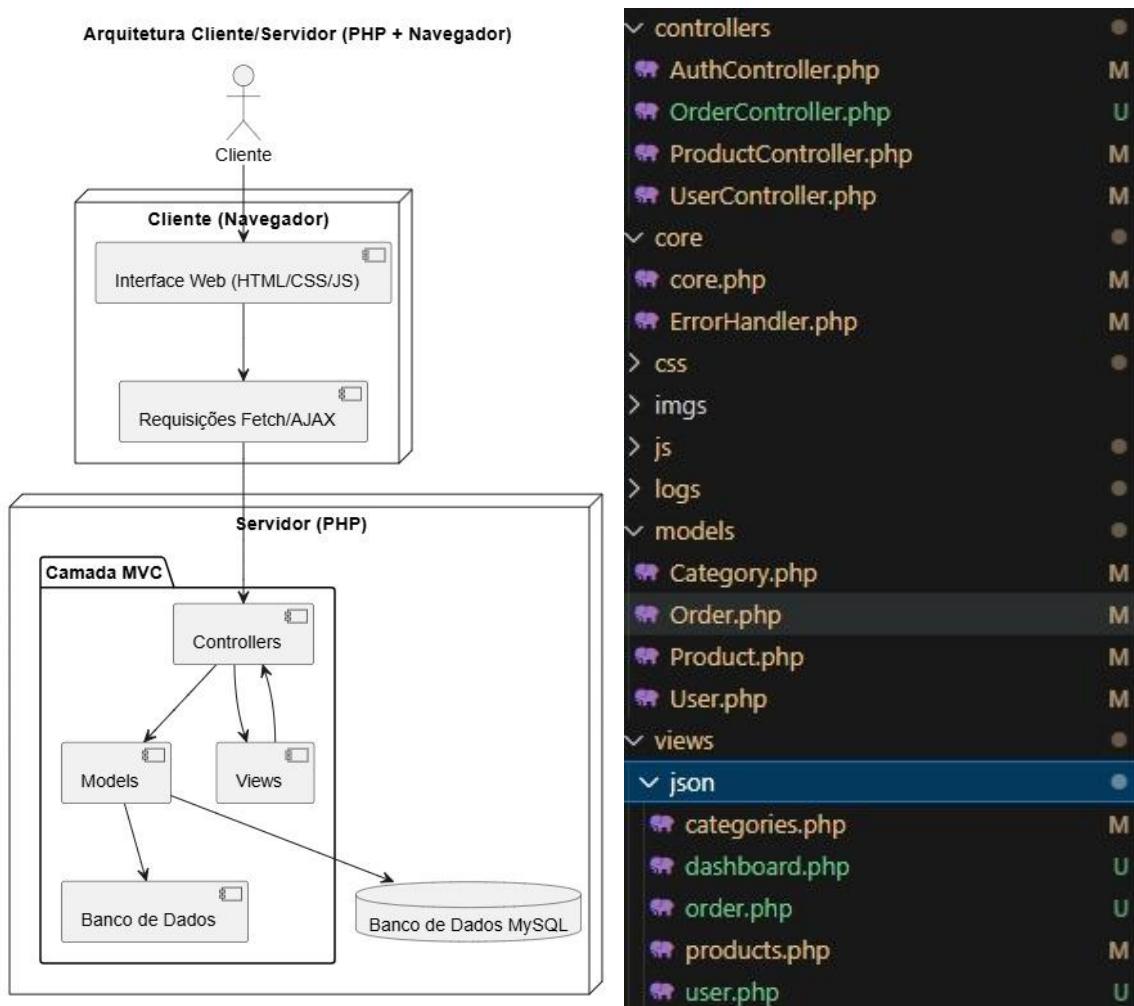
**Model** (regras e dados)

**View** (interface com o usuário)

**Controller** (lógica de controle).

No projeto, essa estrutura foi usada para separar as páginas do site (View), as regras de negócio em PHP (Controller) e as tabelas do banco de dados (Model), facilitando a manutenção e organização do código.

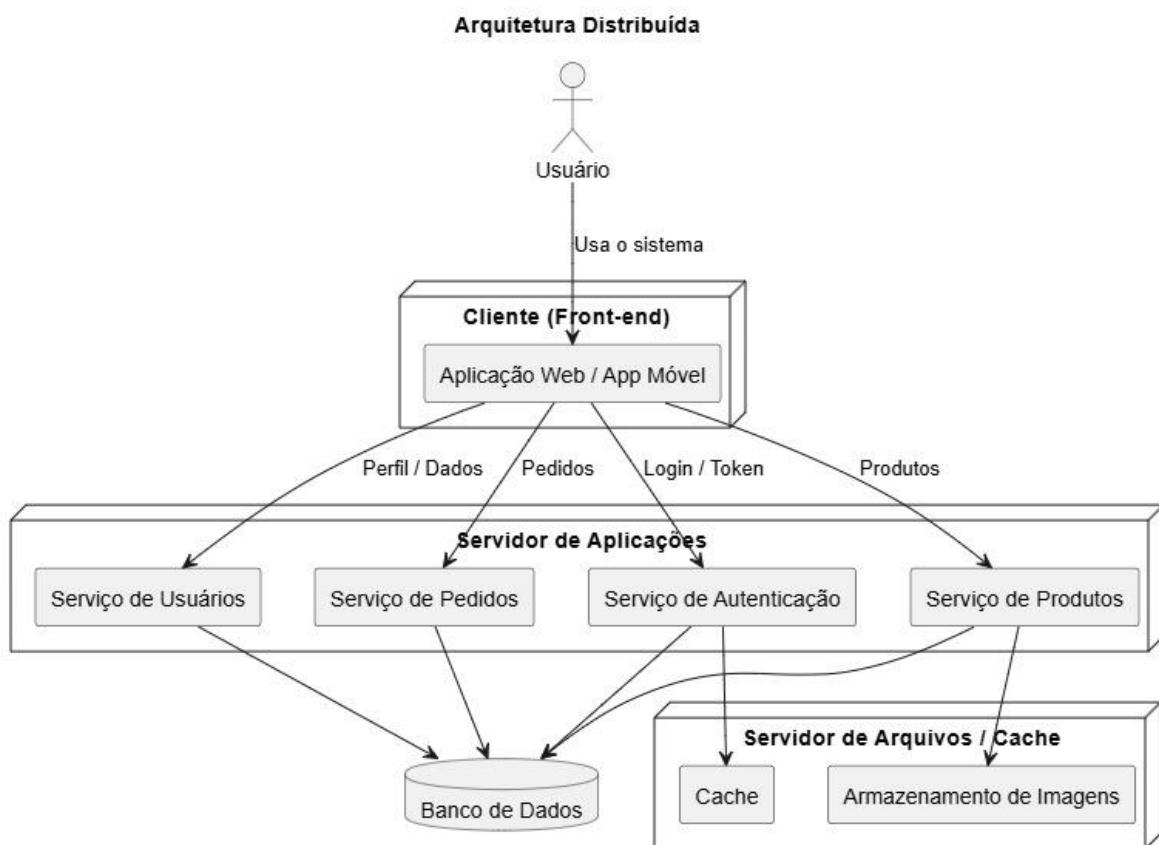
Figura x – Arquitetura MVC



Fonte: Elaborado pelo autor (2025).

### 3.2 Arquitetura distribuída

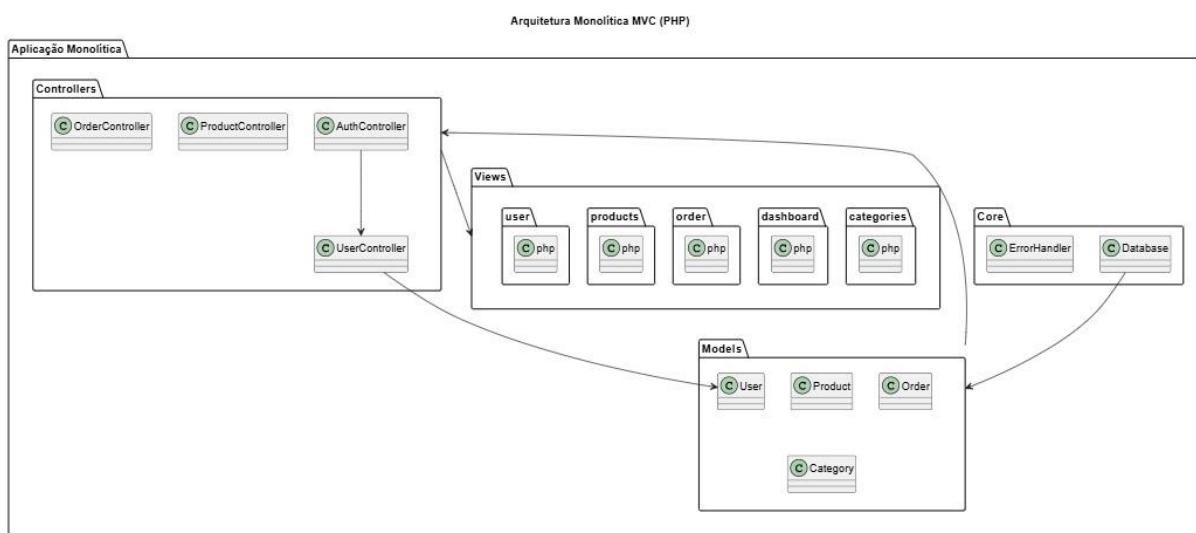
Figura x – Arquitetura Distribuída



Fonte: Elaborado pelo autor (2025).

### 3.3 Arquitetura monolito

Figura x – Arquitetura Monolito



Fonte: Elaborado pelo autor (2025).

## 4. REQUISITOS DO SOFTWARE

### 4.1 Requisitos não funcionais

Tabela 1 – Requisitos Funcionais

Código	Descrição
RF01	O sistema deve gerar automaticamente uma mensagem pronta com os detalhes do pedido e enviá-la para o número de WhatsApp da pastelaria após a finalização da compra.
RF02	O sistema deve permitir o cadastro de clientes com nome, telefone e senha.
RF03	O sistema deve permitir o cadastro de produtos com nome, descrição, preço e imagem.
RF04	O cliente deve conseguir visualizar o catálogo de produtos disponíveis.
RF05	O cliente deve poder adicionar produtos ao carrinho de compras.
RF06	O sistema deve permitir a finalização do pedido.

Fonte: Elaborado pelo autor (2025).

## 4.2 Requisitos não funcionais

Tabela 2 – Requisitos Não Funcionais

Código	Descrição
RNF01	O sistema deve utilizar um banco de dados relacional (MySQL ou SQLite).
RNF02	O sistema deve apresentar uma interface simples e intuitiva para o usuário.
RNF03	O sistema deve funcionar em navegador web.
RNF04	O tempo de resposta para consultas deve ser de até 2 segundos.
RNF05	O sistema deve seguir boas práticas de segurança no processo de login e armazenamento de senhas.

Fonte: Elaborado pelo autor (2025).

## 5 DIAGRAMAS

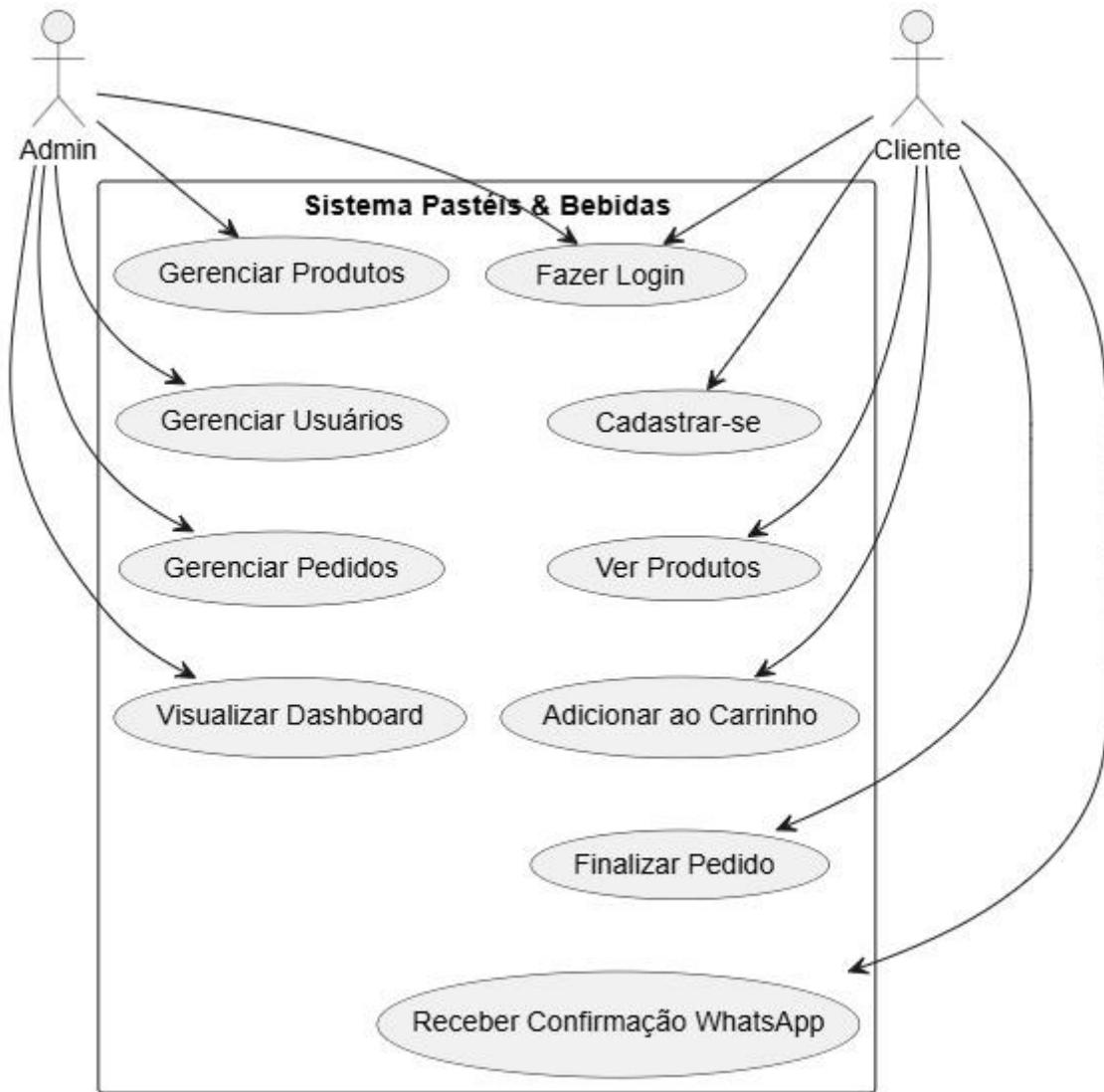
### 4.1 Diagrama de Casos de Uso

Atores:

- **Cliente** (realiza cadastro, visualiza produtos, faz pedidos)

- Administrador (gerência produtos e pedido, visualiza dashboard)

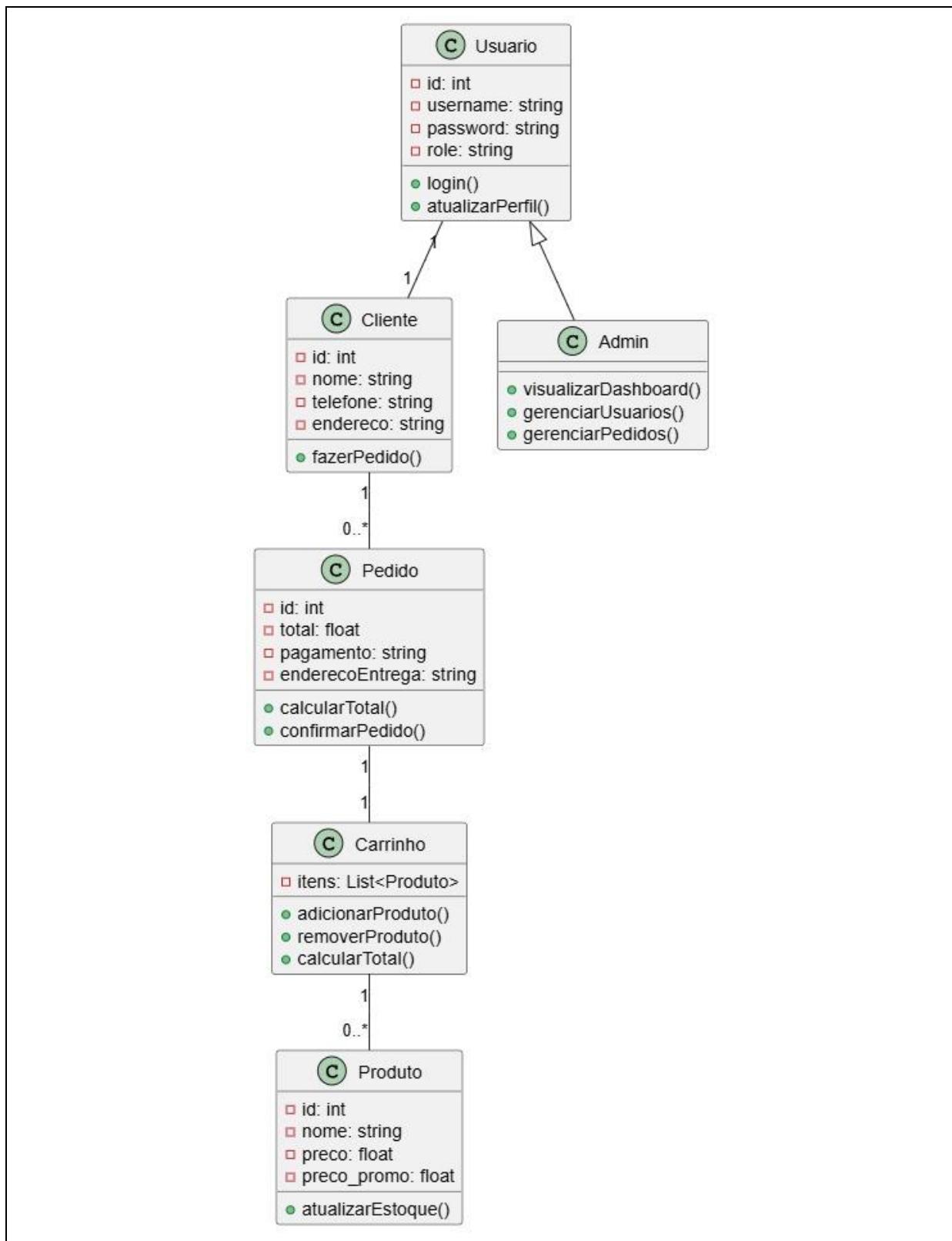
Figura x – Diagrama casos de uso



**Fonte:** Elaborado pelo autor (2025).

#### 4.2 Diagrama de classes

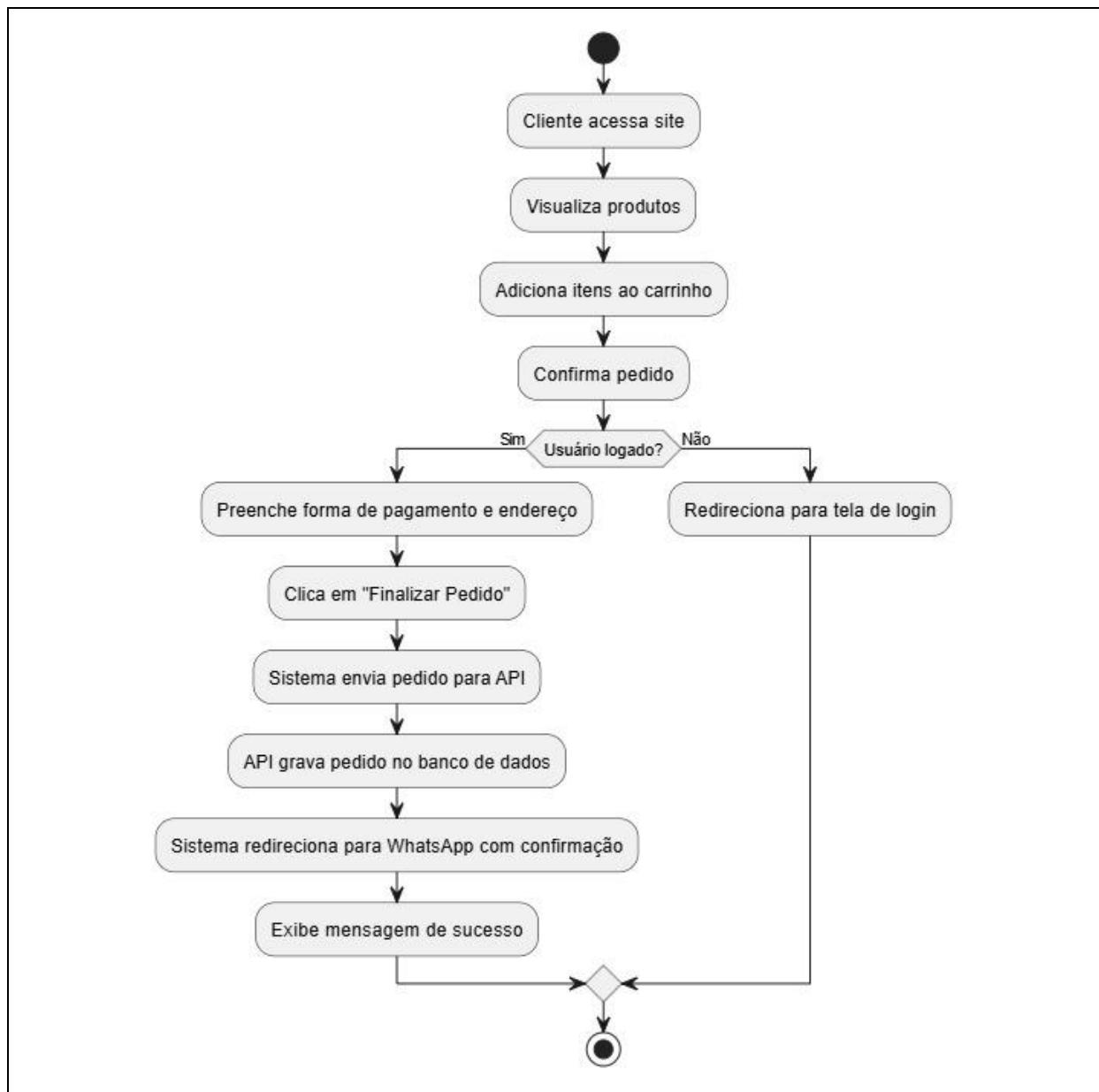
Figura x – Diagrama de classes



**Fonte:** Elaborado pelo autor (2025).

#### 4.3 Diagrama de atividades

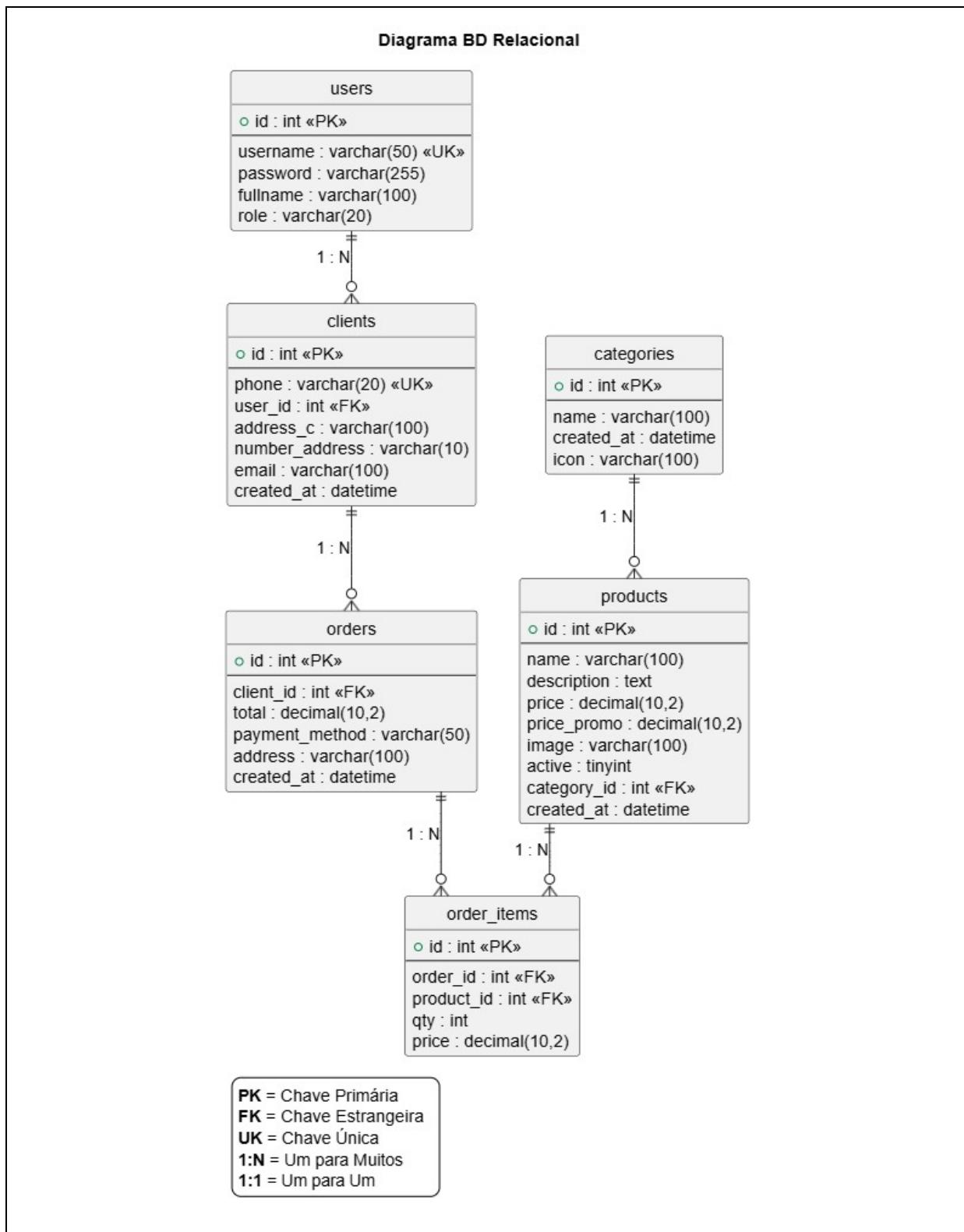
Figura x – Diagrama de atividades



**Fonte:** Elaborado pelo autor (2025).

## 5. DIAGRAMA RELACIONAL DO BANCO DE DADOS

Figura x – Diagrama relacional do banco de dados



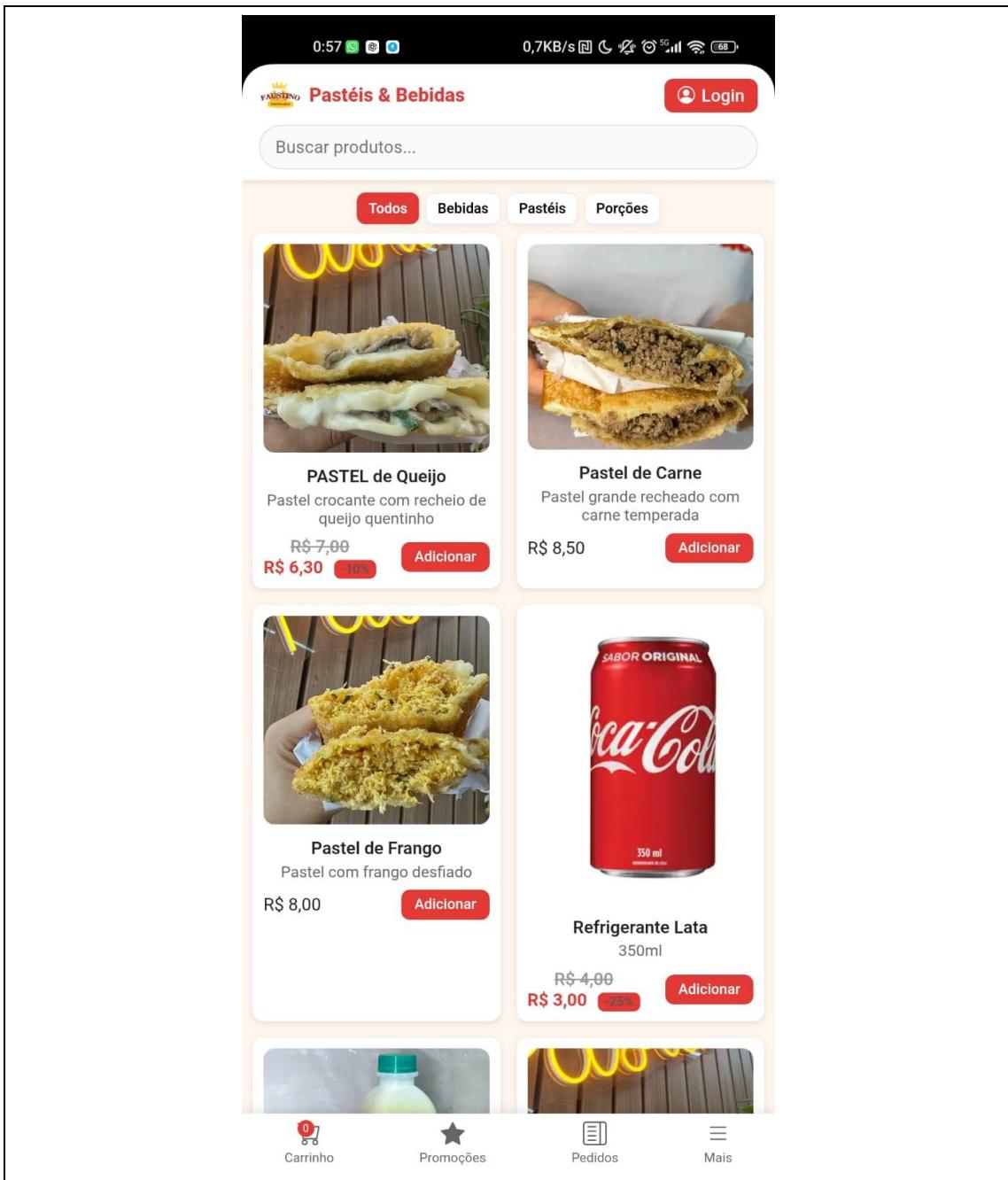
**Fonte:** Elaborado pelo autor (2025).

## 6 PROTÓTIPO DAS TELAS

### 6.1 Tela de Catálogo de Produtos

A Imagem 1, apresenta todos os produtos cadastrados no sistema, exibindo nome, descrição, preço e imagem. Permite ao cliente visualizar as opções disponíveis e selecionar itens para compra.

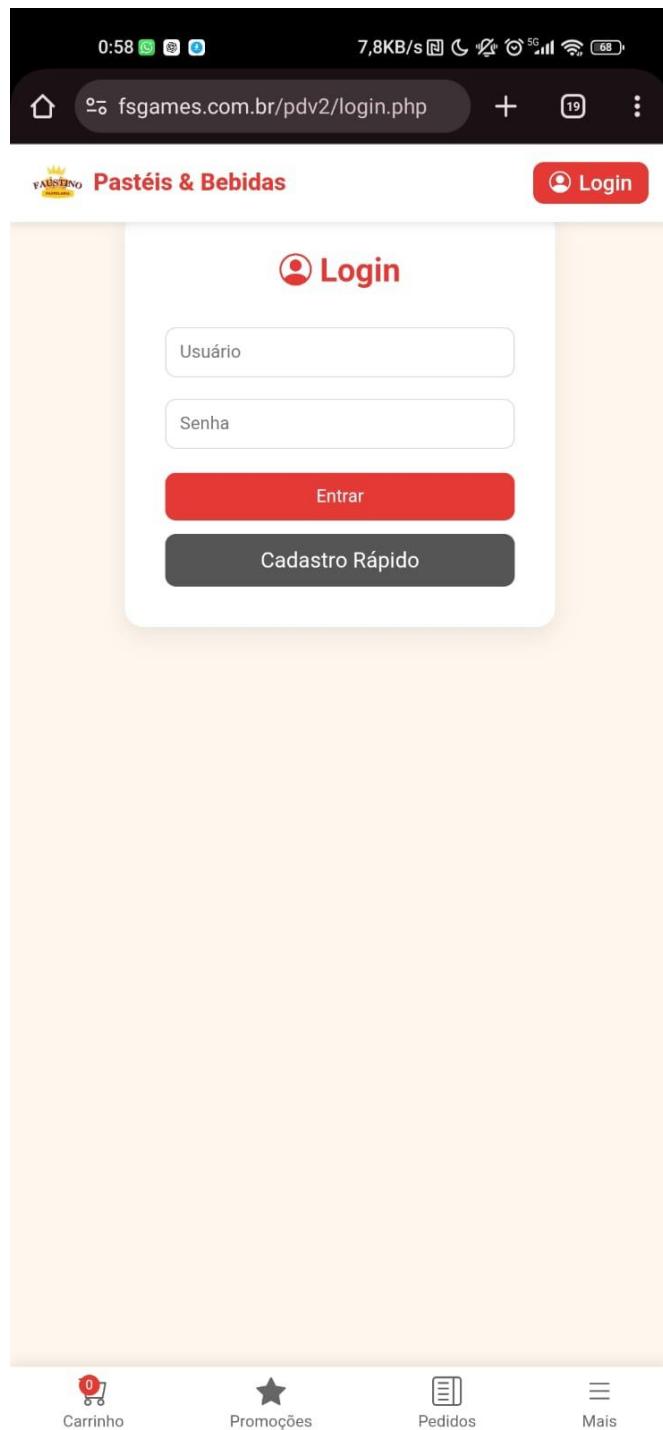
Imagen x – Tela Inicial



Fonte: Elaborado pelo autor (2025).

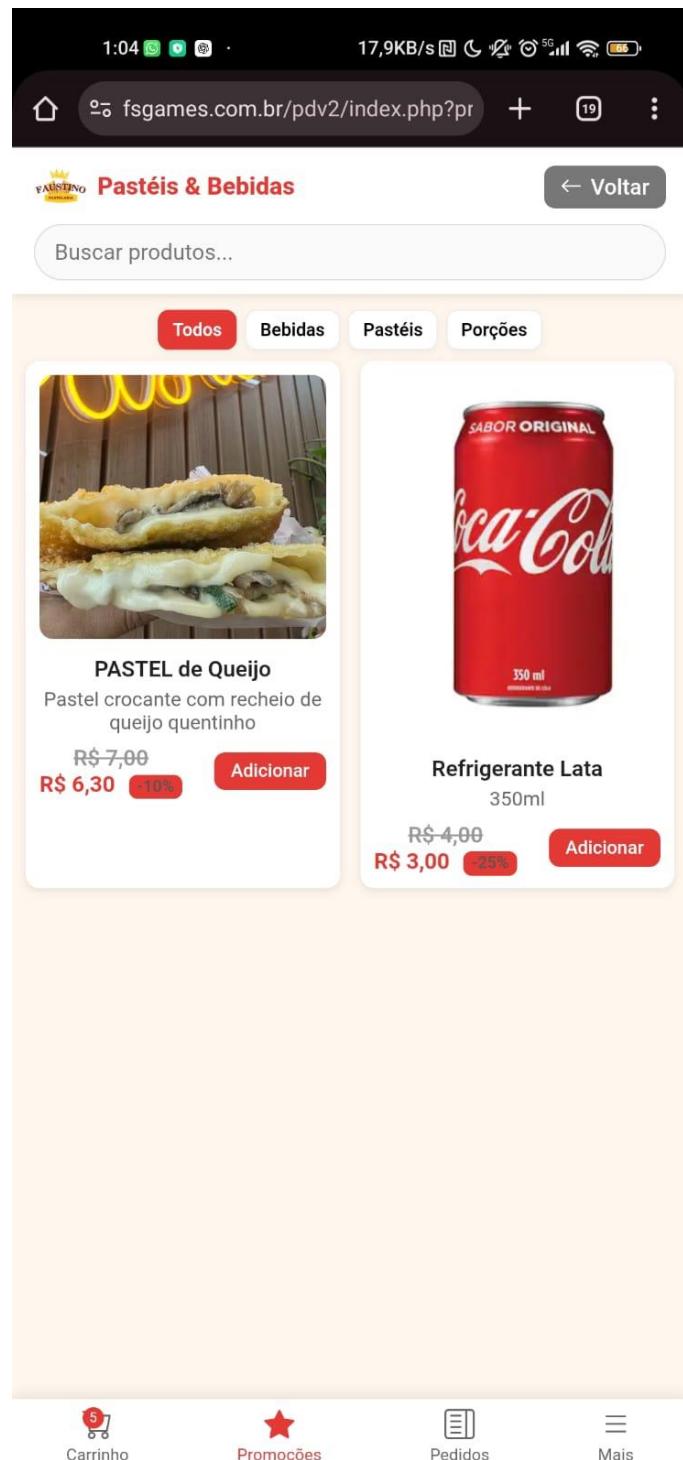
## 6.2 Tela de Login

Permite o acesso de clientes e administradores ao sistema, solicitando nome de usuário e senha. Garante segurança e controle de acesso às funções do sistema.



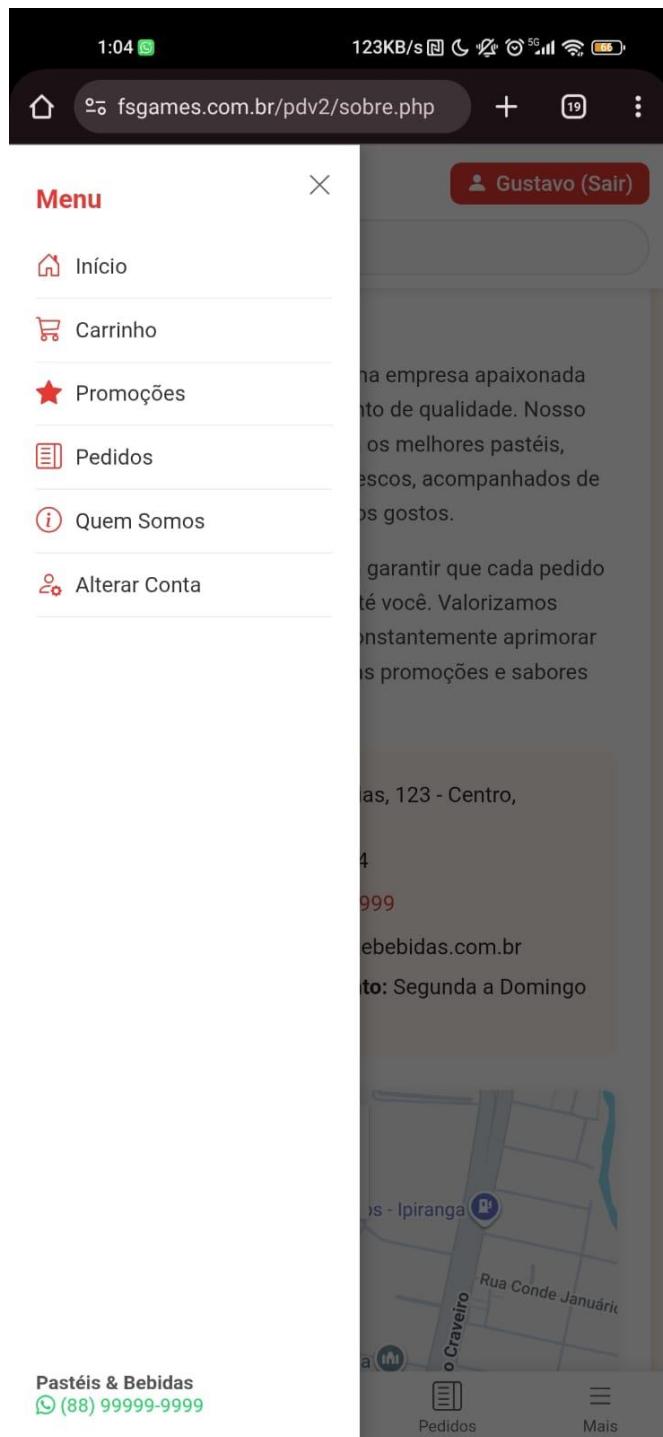
### 6.3 Tela de promoções

Exibe produtos com descontos e ofertas especiais, destacando visualmente os preços promocionais para incentivar as vendas.



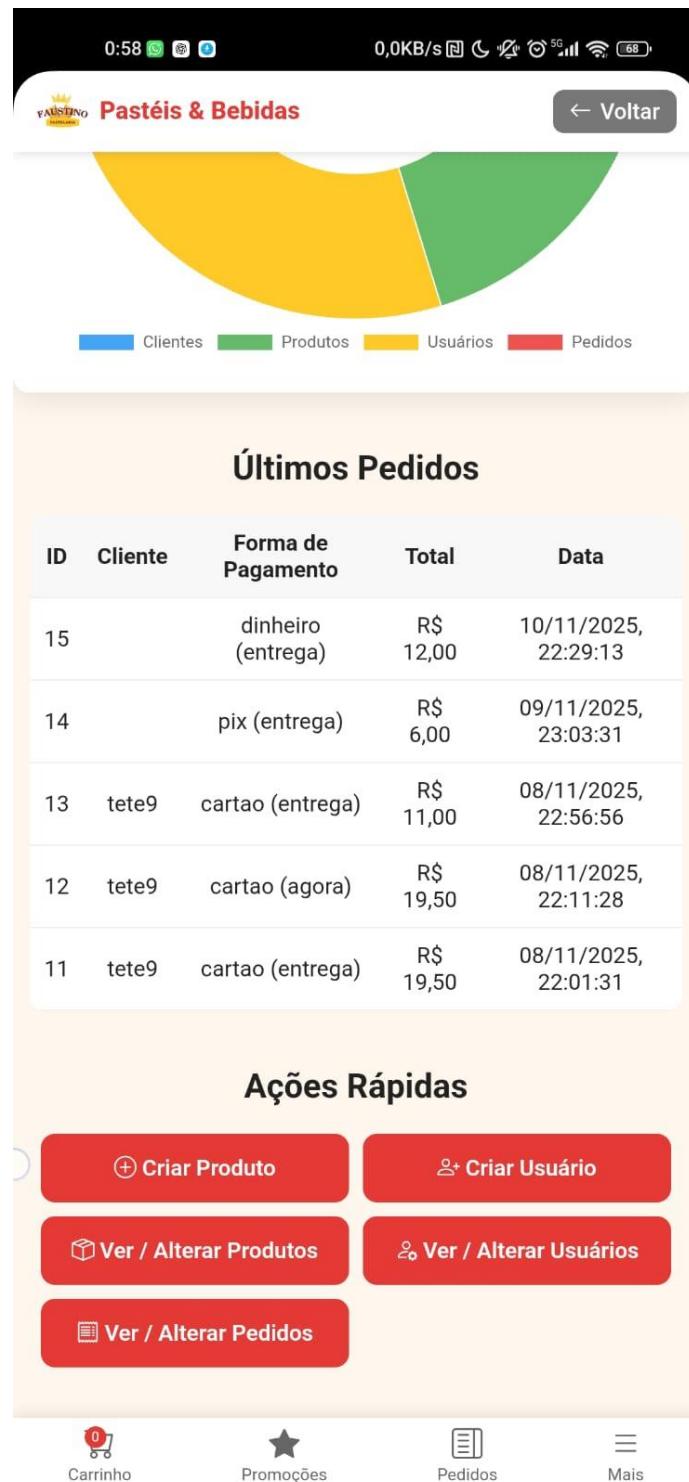
#### 6.4 Menu cliente

Apresenta as principais funções disponíveis para o cliente, como visualizar produtos, acessar o carrinho, verificar pedidos e editar dados pessoais.



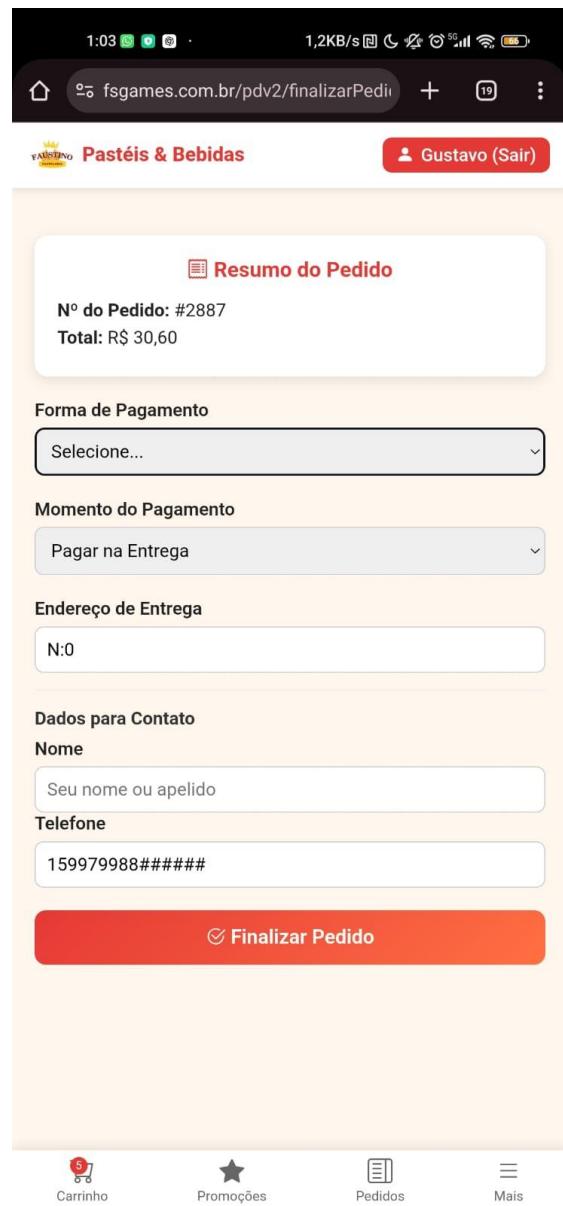
## 6.5 Funções administrador

Disponibiliza ferramentas para o administrador gerenciar produtos, clientes e pedidos. Inclui opções de cadastro, edição e exclusão de registros.



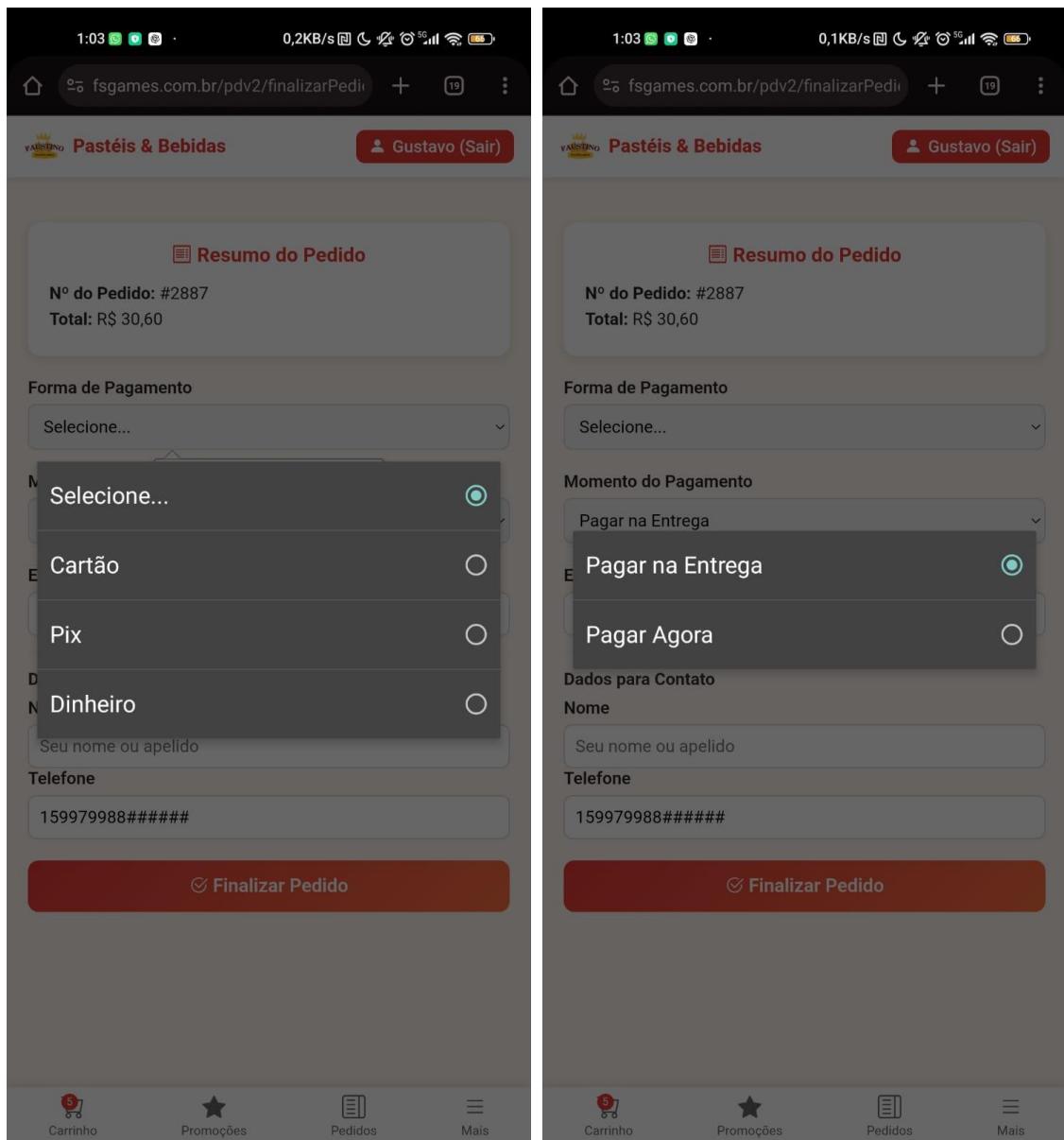
## 6.6 Tela de finalizar produto

Mostra o resumo dos itens selecionados, valor total e formas de pagamento. Ao confirmar, o sistema registra o pedido e envia automaticamente uma mensagem pronta com os detalhes para o WhatsApp da pastelaria.



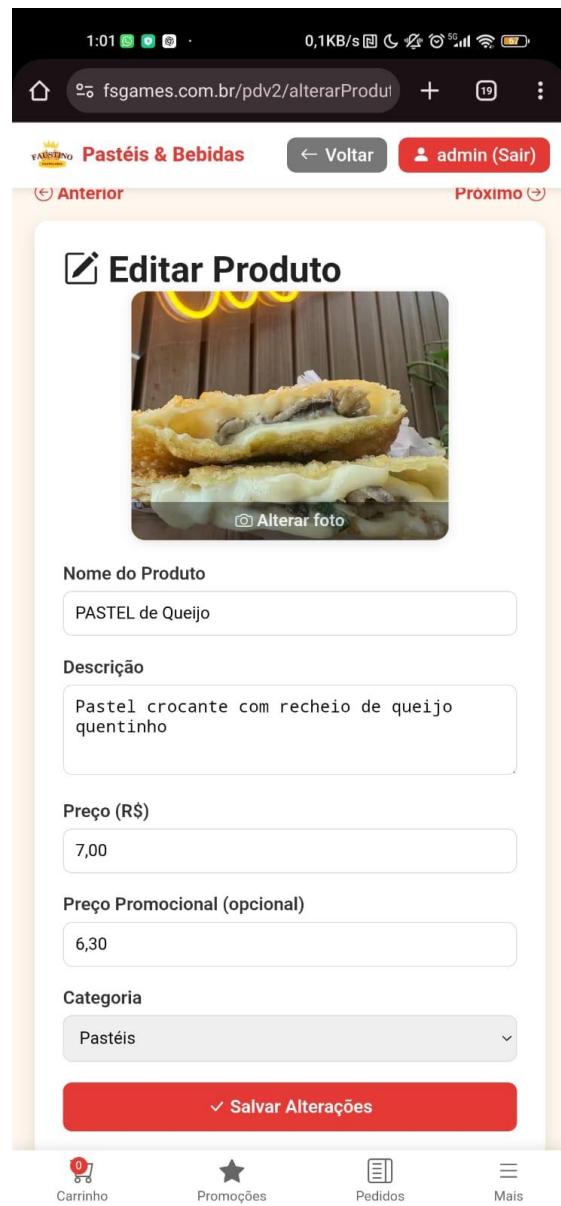
## 6.7 Submenus da tela de finalizar produto

Permitem que o cliente revise endereço, forma de pagamento e confirme os dados antes de concluir o pedido.



## 6.8 Editar produto

Utilizada pelo administrador para alterar informações de produtos já cadastrados, como preço, descrição ou imagem.



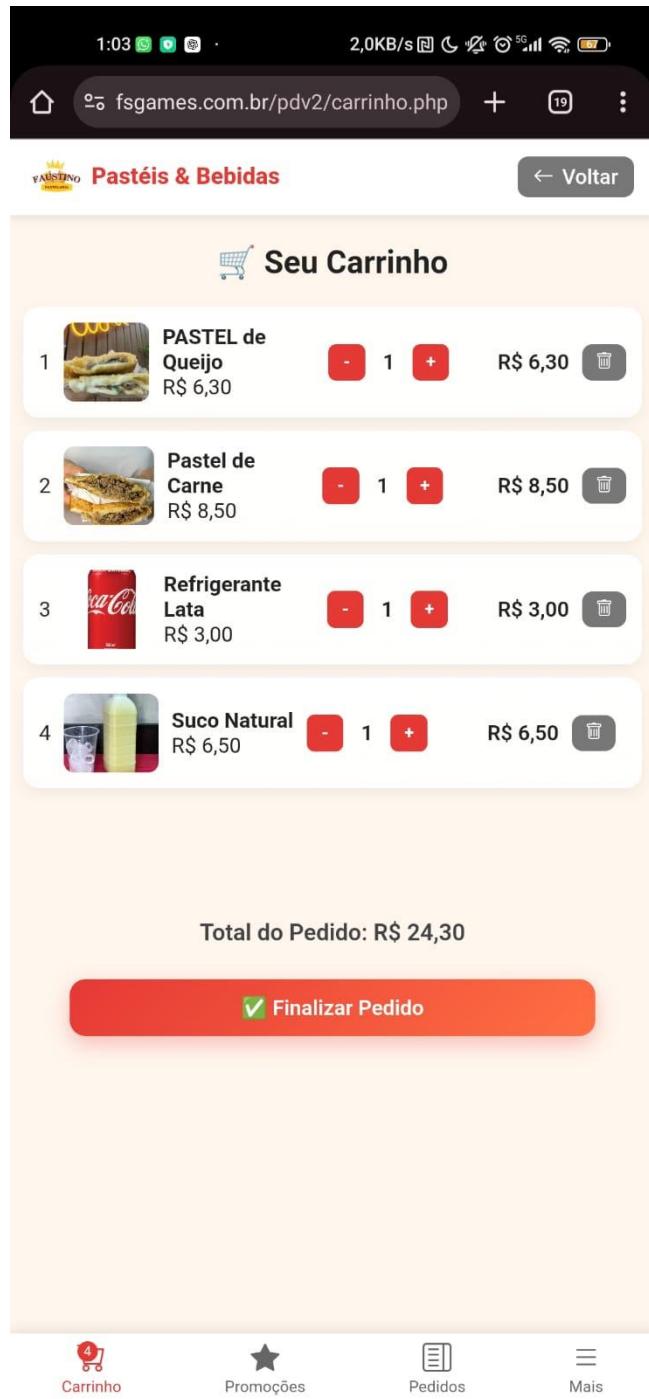
## 6.9 Editar pedido

Permite ao administrador visualizar pedidos realizados e fazer ajustes em casos de erro ou alteração solicitada pelo cliente.



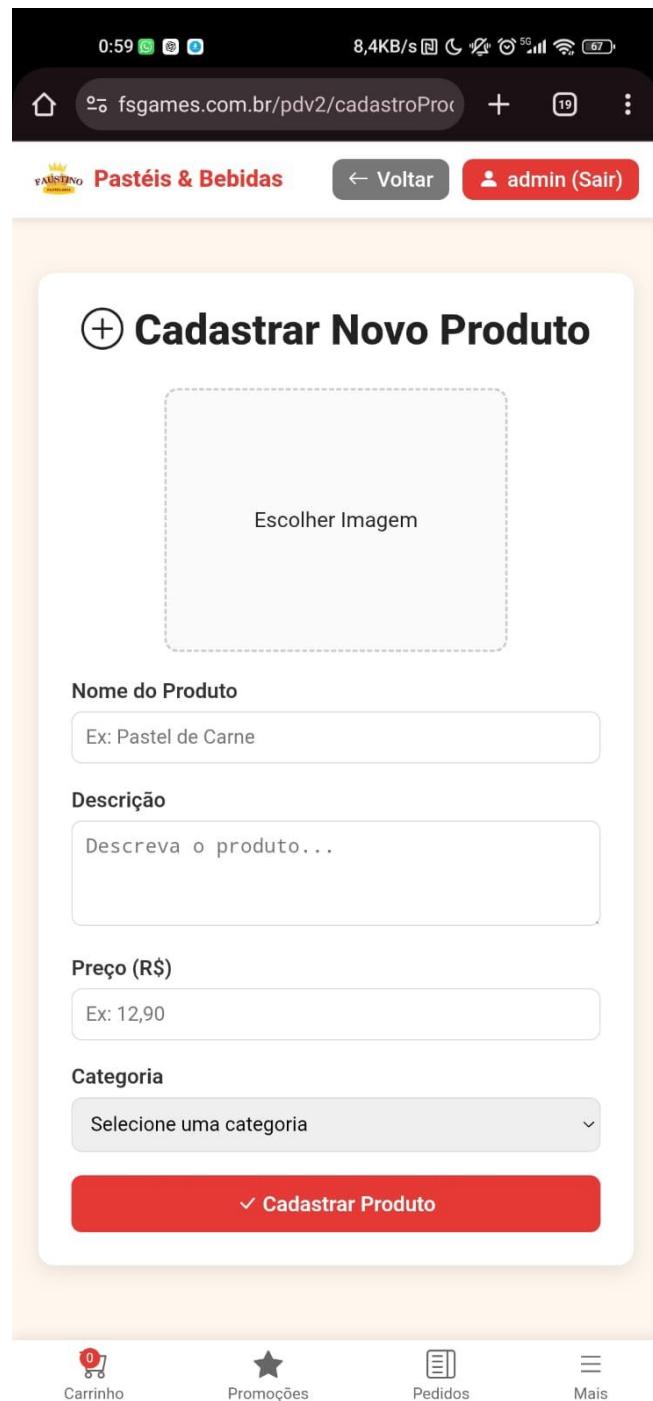
## 6.10 Tela de carrinho de pedidos

Mostra os produtos adicionados ao carrinho, quantidades, valores individuais e total da compra. O cliente pode remover itens ou seguir para a finalização.



## 6.11 Cadastrar novo produto

Permite que o administrador adicione novos produtos, informando nome, descrição, preço, categoria e imagem ilustrativa.



## 6.12 Tela de Cadastro de Usuário

Usada por novos clientes para criar uma conta no sistema, preenchendo dados básicos como nome, telefone, e-mail e senha.

The screenshot shows a mobile application interface for user registration. At the top, there is a header bar with the URL [fsgames.com.br/pdv2/cadastroClie](http://fsgames.com.br/pdv2/cadastroClie). Below the header, the title "Pastéis & Bebidas" is displayed next to a crown icon. On the right side of the header are buttons for "Voltar" (Back) and "admin (Sair)" (Logout). The main content area is titled "Cadastro Rápido" (Quick Registration) with a plus sign icon. It contains seven input fields: "Usuário", "Senha", "Nome completo", "Telefone", "Endereço", "Número", and "E-mail". A large red button at the bottom right labeled "Cadastrar" (Register) is used to submit the form. At the very bottom of the screen, there are four navigation icons: "Carrinho" (Cart) with a red notification badge showing '0', "Promoções" (Promotions), "Pedidos" (Orders), and "Mais" (More).

## Cadastro Rápido

Usuário

Senha

Nome completo

Telefone

Endereço

Número

E-mail

Cadastrar

0  
7

Carrinho



Promoções



Pedidos



Mais