



Universidade Federal  
de Ouro Preto

**Universidade Federal de Ouro Preto  
Instituto de Ciências Exatas e Aplicadas  
Departamento de Computação e Sistemas**

# **Sistema de Controle Gerencial para Microempreendedores Individuais**

**André Lucas Santos**

## **TRABALHO DE CONCLUSÃO DE CURSO**

**ORIENTAÇÃO:**  
Fernando Bernardes de Oliveira

**Março, 2026  
João Monlevade—MG**

**André Lucas Santos**

# **Sistema de Controle Gerencial para Microempreendedores Individuais**

Orientador: Fernando Bernardes de Oliveira

Monografia apresentada ao curso de Sistemas de Informação do Instituto de Ciências Exatas e Aplicadas, da Universidade Federal de Ouro Preto, como requisito parcial para aprovação na Disciplina “Trabalho de Conclusão de Curso II”.

**Universidade Federal de Ouro Preto**

**João Monlevade**

**Março de 2026**

## SISBIN - SISTEMA DE BIBLIOTECAS E INFORMAÇÃO

S237s Santos, Andre Lucas.

Sistema de controle gerencial para microempreendedores individuais.  
[manuscrito] / Andre Lucas Santos. - 2026.  
66 f.: il.: color., tab..

Orientador: Prof. Dr. Fernando Bernardes Oliveira.

Monografia (Bacharelado). Universidade Federal de Ouro Preto.  
Instituto de Ciências Exatas e Aplicadas. Graduação em Sistemas de  
Informação .

1. Aplicações Web - Desenvolvimento. 2. Controle de estoque. 3.  
Inteligência competitiva (Administração). 4. Pequenas e médias  
empresas - Administração. 5. Sistemas de informação gerencial. 6.  
Sucesso nos negócios. I. Oliveira, Fernando Bernardes. II. Universidade  
Federal de Ouro Preto. III. Título.

CDU 658.017.3:004.775

Bibliotecário(a) Responsável: Flavia Reis - CRB6/2431



## FOLHA DE APROVAÇÃO

**André Lucas Santos**

### **Sistema de Controle Gerencial para Microempreendedores Individuais**

Monografia apresentada ao Curso de Sistemas de Informação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Sistemas de Informação

Aprovada em 03 de março de 2026

#### Membros da banca

Prof. Dr. Fernando Bernardes de Oliveira - Orientador (Universidade Federal de Ouro Preto)  
Profª. Drª. Gilda Aparecida de Assis - Avaliadora (Universidade Federal de Ouro Preto)  
Profª. Drª. Helen de Cássia Sousa Da Costa Lima - Avaliadora (Universidade Federal de Ouro Preto)

Fernando Bernardes de Oliveira, orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 12/03/2026



Documento assinado eletronicamente por **Fernando Bernardes de Oliveira, PROFESSOR DE MAGISTERIO SUPERIOR**, em 12/03/2026, às 16:06, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site [http://sei.ufop.br/sei/controlador\\_externo.php?acao=documento\\_conferir&id\\_orgao\\_acesso\\_externo=0](http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0), informando o código verificador **1074428** e o código CRC **5C903FB7**.

*Este trabalho é dedicado a Deus, a Ele toda a glória.*

# Agradecimentos

Primeiramente gostaria de agradecer a Deus, que esteve comigo em toda a jornada. Agradeço a minha família, pelo apoio e por tudo que fizeram por mim até aqui. Sou grato pelas amizades construídas na faculdade, que me ajudaram a trilhar meu caminho. Sou grato pelo meu orientador Fernando Bernardes, pela sua confiança e auxílio nas dificuldades. Por fim, sou grato a Universidade Federal de Ouro Preto, pelo ensino de qualidade e pela oportunidade de crescimento.

# Resumo

Este trabalho apresenta o desenvolvimento e a validação de uma plataforma web de gestão voltada para o Microempreendedor Individual (MEI). O problema central abordado é a complexidade dos *Enterprise Resource Planning* (ERP) tradicionais, que frequentemente dificultam a tomada de decisão estratégica por pequenos empreendedores devido ao alto custo cognitivo. O objetivo deste estudo foi criar e validar uma ferramenta simplificada que integra funcionalidades essenciais de controle de estoque e vendas a módulos de *Business Intelligence* (BI). A metodologia baseou-se no uso do *framework* Laravel para o *backend* e React com Material UI (MUI) para a interface de utilizador. A validação do sistema ocorreu em três frentes: testes automatizados de integridade de dados, auditorias de desempenho via Google Lighthouse e testes de usabilidade com o público-alvo utilizando a escala Satisfação e Usabilidade (SUS). Os resultados demonstraram que o sistema é tecnicamente robusto e intuitivo, oferecendo clareza financeira através de métricas como a Curva Classificação de Inventário (Princípio de Pareto) (ABC) e análises de tendência por regressão linear. Conclui-se que a solução cumpre o propósito de democratizar o acesso a ferramentas de gestão eficientes e acessíveis para o MEI.

**Palavras-chave:** Microempreendedor Individual. Gestão de Negócios. Laravel. React. Business Intelligence.

# Abstract

This work presents the development and validation of a web management platform tailored for the [MEI](#). The central problem addressed is the complexity of traditional [ERP](#) systems, which often hinder strategic decision-making for small entrepreneurs due to high cognitive costs. The objective of this study was to create and validate a simplified tool that integrates essential inventory and sales control functionalities with [BI](#) modules. The methodology was based on the use of the Laravel framework for the backend and React with [Mui](#) for the user interface. System validation occurred on three fronts: automated data integrity tests, performance audits via Google Lighthouse, and usability tests with the target audience using the [SUS](#) scale. The results demonstrated that the system is technically robust and intuitive, offering financial clarity through metrics such as the [ABC](#) Curve and trend analysis via linear regression. It is concluded that the solution fulfills the purpose of democratizing access to efficient and affordable management tools for the [MEI](#).

**Keywords:** Individual Micro-entrepreneur. Business Management. Laravel. React. Business Intelligence.

# Lista de ilustrações

|                                                                                        |    |
|----------------------------------------------------------------------------------------|----|
| Figura 1 – Logomarca do sistema. . . . .                                               | 32 |
| Figura 2 – Tela Inicial: Hero Section e Navegação . . . . .                            | 33 |
| Figura 3 – Detalhamento de Funcionalidades e Benefícios . . . . .                      | 34 |
| Figura 4 – Interface de Autenticação com Validação de Erros . . . . .                  | 35 |
| Figura 5 – Formulário de Cadastro com Prevenção de Falhas . . . . .                    | 36 |
| Figura 6 – Painel de Controle com Indicadores de Desempenho . . . . .                  | 37 |
| Figura 7 – Listagem e Gestão de Clientes . . . . .                                     | 38 |
| Figura 8 – Gestão de Catálogo e Estrutura Hierárquica . . . . .                        | 39 |
| Figura 9 – Gestão de Produtos com Filtros Avançados . . . . .                          | 40 |
| Figura 10 – Ações Rápidas e Movimentação de Estoque . . . . .                          | 41 |
| Figura 11 – Histórico de Vendas e Ações Operacionais . . . . .                         | 42 |
| Figura 12 – Interface de Ponto de Venda (PDV): Catálogo e Validação de Saldo . . . . . | 43 |
| Figura 13 – Visão Geral e Indicadores Macroeconômicos . . . . .                        | 45 |
| Figura 14 – Visão Específica (alta) . . . . .                                          | 47 |
| Figura 15 – Visão Específica (queda) . . . . .                                         | 48 |
| Figura 16 – Botão de Acesso à Ajuda (FAB) . . . . .                                    | 49 |
| Figura 17 – Interface do Guia Interativo Passo a Passo . . . . .                       | 50 |
| Figura 18 – Adaptação da Interface para Dispositivos Móveis . . . . .                  | 51 |
| Figura 19 – Relatório de execução dos testes automatizados . . . . .                   | 54 |
| Figura 20 – Resultado da auditoria Google Lighthouse . . . . .                         | 55 |
| Figura 21 – Nível de familiaridade tecnológica . . . . .                               | 56 |
| Figura 22 – Experiência prévia em gestão . . . . .                                     | 56 |
| Figura 23 – Atratividade visual da interface . . . . .                                 | 56 |
| Figura 24 – Clareza do menu principal . . . . .                                        | 56 |
| Figura 25 – Feedback sobre o Modo Escuro . . . . .                                     | 57 |
| Figura 26 – Compreensão da Árvore Mercadológica . . . . .                              | 57 |
| Figura 27 – Facilidade de cadastro de produtos . . . . .                               | 57 |
| Figura 28 – Fluidez do fluxo de vendas . . . . .                                       | 57 |
| Figura 29 – Utilidade do sistema de ajuda . . . . .                                    | 57 |
| Figura 30 – Desejo de uso frequente . . . . .                                          | 57 |
| Figura 31 – Percepção de complexidade . . . . .                                        | 58 |
| Figura 32 – Facilidade de uso geral . . . . .                                          | 58 |
| Figura 33 – Velocidade de aprendizado . . . . .                                        | 58 |
| Figura 34 – Nível de confiança no uso . . . . .                                        | 58 |
| Figura 35 – Funcionalidades favoritas . . . . .                                        | 58 |
| Figura 36 – Relatos de erros ou bugs . . . . .                                         | 58 |

|                                                               |    |
|---------------------------------------------------------------|----|
| Figura 37 – Sugestões de melhorias . . . . .                  | 59 |
| Figura 38 – Diagrama de Estrutura do Banco de Dados . . . . . | 65 |

# Lista de tabelas

|                                                                                   |    |
|-----------------------------------------------------------------------------------|----|
| Tabela 1 – Comparativo entre a solução proposta e as ferramentas existentes . . . | 23 |
|-----------------------------------------------------------------------------------|----|

# Lista de abreviaturas e siglas

|                         |                                                    |
|-------------------------|----------------------------------------------------|
| <b>ABC</b>              | Classificação de Inventário (Princípio de Pareto)  |
| <b>API</b>              | <i>Application Programming Interface</i>           |
| <b>ARIA</b>             | <i>Accessible Rich Internet Applications</i>       |
| <b>BI</b>               | <i>Business Intelligence</i>                       |
| <b>CMV</b>              | Custo de Mercadoria Vendida                        |
| <b>CSR</b>              | <i>Client-Side Rendering</i>                       |
| <b>DW</b>               | <i>Data Warehouse</i>                              |
| <b>DER</b>              | Diagrama Entidade-Relacionamento                   |
| <b>DOM</b>              | <i>Document Object Model</i>                       |
| <b>EAN</b>              | <i>European Article Number</i>                     |
| <b>ERP</b>              | <i>Enterprise Resource Planning</i>                |
| <b>FAB</b>              | <i>Floating Action Button</i>                      |
| <b>GPC</b>              | <i>Global Product Classification</i>               |
| <b>HTML</b>             | <i>Hypertext Markup Language</i>                   |
| <b>ICMS</b>             | Imposto sobre Circulação de Mercadorias e Serviços |
| <b>ID</b>               | <i>Identifier</i>                                  |
| <b>JSON</b>             | <i>JavaScript Object Notation</i>                  |
| <b>KPI</b>              | <i>Key Performance Indicator</i>                   |
| <b>MEI</b>              | Microempreendedor Individual                       |
| <b>MUI</b>              | Material UI                                        |
| <b>MVC</b>              | <i>Model-View-Controller</i>                       |
| <b>ORM</b>              | <i>Object-Relational Mapping</i>                   |
| <b>PDV</b>              | Ponto de Venda                                     |
| <b><math>R^2</math></b> | Coefficiente de Determinação                       |
| <b>REST</b>             | <i>Representational State Transfer</i>             |
| <b>RWD</b>              | Design Responsivo e Mobilidade                     |
| <b>SaaS</b>             | <i>Software as a Service</i>                       |
| <b>SEO</b>              | <i>Search Engine Optimization</i>                  |
| <b>SGBD</b>             | Sistema de Gerenciamento de Banco de Dados         |

|            |                                  |
|------------|----------------------------------|
| <b>SKU</b> | <i>Stock Keeping Unit</i>        |
| <b>SPA</b> | <i>Single Page Application</i>   |
| <b>SQL</b> | <i>Structured Query Language</i> |
| <b>SUS</b> | Satisfação e Usabilidade         |
| <b>TDD</b> | <i>Test-Driven Development</i>   |
| <b>UI</b>  | <i>User Interface</i>            |
| <b>URL</b> | <i>Uniform Resource Locator</i>  |

# Sumário

|          |                                                       |           |
|----------|-------------------------------------------------------|-----------|
| <b>1</b> | <b>INTRODUÇÃO</b>                                     | <b>16</b> |
| 1.1      | O problema de pesquisa                                | 16        |
| 1.2      | Objetivos                                             | 17        |
| 1.3      | Metodologia                                           | 17        |
| 1.4      | Organização do trabalho                               | 18        |
| <b>2</b> | <b>FUNDAMENTAÇÃO TEÓRICA</b>                          | <b>19</b> |
| 2.1      | Padrões de Arquitetura e Organização de Dados         | 19        |
| 2.2      | Evolução das Interfaces: Monólitos e SPAs             | 19        |
| 2.3      | Desenvolvimento Frontend e Tipagem Estática           | 20        |
| 2.4      | Gestão de Estoque e o Princípio de Pareto             | 20        |
| 2.5      | Análise Preditiva e Tendências                        | 20        |
| 2.6      | Arquiteturas Multi-inquilino e Computação em Nuvem    | 21        |
| 2.7      | Sistemas de Apoio à Decisão e Inteligência de Negócio | 21        |
| 2.8      | Qualidade de Software e Verificação Contínua          | 21        |
| 2.9      | Trabalhos Relacionados e Soluções de Mercado          | 21        |
| 2.9.1    | Perspectiva Acadêmica sobre a Informatização do MEI   | 22        |
| 2.9.2    | Análise de Soluções de Mercado                        | 22        |
| 2.9.3    | Posicionamento do Sistema Proposto                    | 22        |
| 2.10     | Considerações finais                                  | 23        |
| <b>3</b> | <b>DESENVOLVIMENTO E IMPLEMENTAÇÃO</b>                | <b>24</b> |
| 3.1      | Modelagem da Base de Dados e Persistência             | 24        |
| 3.1.1    | Isolamento de Dados ( <i>Multitenancy</i> )           | 24        |
| 3.1.2    | Estrutura de Catálogo: Árvore Mercadológica           | 25        |
| 3.1.3    | Rastreabilidade Financeira e Integridade de Estoque   | 25        |
| 3.1.4    | Auditoria e Padrão Observer                           | 26        |
| 3.2      | Processamento de Dados e Inteligência de Negócio      | 26        |
| 3.2.1    | Otimização de Performance com Query Builder           | 26        |
| 3.2.2    | Implementação do Algoritmo de Curva ABC               | 27        |
| 3.2.3    | Algoritmo de Tendência: Regressão Linear Simples      | 28        |
| 3.3      | Visualização de Dados e Interface                     | 28        |
| 3.3.1    | Renderização de Gráficos e Componentização            | 28        |
| 3.3.2    | Estratégia de Carregamento Assíncrono e Skeleton UI   | 29        |
| 3.4      | Tecnologias e Ferramentas Utilizadas                  | 29        |
| 3.4.1    | Ecossistema de Backend e Persistência                 | 29        |

|         |                                                                               |    |
|---------|-------------------------------------------------------------------------------|----|
| 3.4.2   | Ecosistema de Frontend e Interface . . . . .                                  | 30 |
| 3.4.3   | Ferramentas de Engenharia e Qualidade . . . . .                               | 30 |
| 3.5     | <b>Considerações finais</b> . . . . .                                         | 31 |
| 4       | <b>RESULTADOS E DISCUSSÃO</b> . . . . .                                       | 32 |
| 4.1     | <b>Apresentação das Funcionalidades e Interfaces</b> . . . . .                | 32 |
| 4.1.1   | Página Inicial e Proposta de Valor . . . . .                                  | 33 |
| 4.1.2   | Módulo de Autenticação e Registro . . . . .                                   | 34 |
| 4.1.3   | Painel de Controle e Indicadores (Dashboard) . . . . .                        | 36 |
| 4.1.4   | Gestão de Clientes . . . . .                                                  | 37 |
| 4.1.5   | Organização Mercadológica (Árvore e Listagem) . . . . .                       | 38 |
| 4.1.6   | Produtos e Controle de Estoque . . . . .                                      | 39 |
| 4.1.7   | Ponto de Venda (PDV) e Histórico . . . . .                                    | 41 |
| 4.1.8   | Módulo de Estatísticas Avançadas e Inteligência de Negócio . . . . .          | 43 |
| 4.1.9   | Visão Geral: Diagnóstico Macroeconômico . . . . .                             | 43 |
| 4.1.9.1 | Visão Específica (Inteligência de Produto) . . . . .                          | 45 |
| 4.1.9.2 | Métricas Financeiras e Logísticas . . . . .                                   | 45 |
| 4.1.9.3 | Análise de Tendência e Ciclo de Vida . . . . .                                | 46 |
| 4.1.10  | Módulo de Onboarding e Ajuda Contextual . . . . .                             | 49 |
| 4.1.11  | Design Responsivo e Mobilidade Design Responsivo e Mobilidade (RWD) . . . . . | 50 |
| 4.2     | <b>Testes Automatizados de Backend</b> . . . . .                              | 51 |
| 4.2.1   | Integridade do Estoque . . . . .                                              | 52 |
| 4.2.2   | Fluxo de Vendas . . . . .                                                     | 52 |
| 4.2.3   | Segurança e Isolamento de Dados . . . . .                                     | 53 |
| 4.2.4   | Conclusão dos Testes de Backend . . . . .                                     | 53 |
| 4.3     | <b>Análise de Desempenho e Qualidade (Frontend)</b> . . . . .                 | 54 |
| 4.4     | <b>Avaliação da Experiência do Usuário (UX)</b> . . . . .                     | 55 |
| 4.4.1   | Resultados e Análise do Feedback . . . . .                                    | 56 |
| 4.5     | <b>Análise Crítica e Limitações do Projeto</b> . . . . .                      | 59 |
| 4.6     | <b>Considerações finais</b> . . . . .                                         | 60 |
| 5       | <b>CONCLUSÃO</b> . . . . .                                                    | 61 |
| 5.1     | <b>Síntese do Trabalho e Alcance dos Objetivos</b> . . . . .                  | 61 |
| 5.2     | <b>Limitações e Análise Crítica</b> . . . . .                                 | 61 |
| 5.3     | <b>Trabalhos Futuros</b> . . . . .                                            | 62 |
|         | <b>REFERÊNCIAS</b> . . . . .                                                  | 63 |

|                                                           |           |
|-----------------------------------------------------------|-----------|
| <b>APÊNDICES</b>                                          | <b>64</b> |
| <b>APÊNDICE A – MODELAGEM DO BANCO DE DADOS . . . . .</b> | <b>65</b> |

# 1 Introdução

O empreendedorismo no Brasil tem demonstrado uma força crescente, consolidando-se como uma alternativa fundamental de geração de renda e ocupação. Esse movimento é impulsionado pela migração de profissionais que deixam o mercado de trabalho tradicional para gerir seus próprios negócios, transformando a dinâmica econômica do país. Segundo dados divulgados pelo Governo Federal [Brasil \(2025\)](#), o Brasil registrou a abertura de mais de 1,4 milhões de novos pequenos negócios apenas no primeiro trimestre de 2025, sendo que 78% destes correspondem a [MEI](#).

Este crescimento expressivo, contudo, vem acompanhado de desafios críticos. A sustentabilidade destes negócios está intrinsecamente ligada à capacidade de gestão dos seus proprietários. [Almeida e Silva \(2024\)](#) destacam que a falta de educação financeira é uma das principais causas da elevada taxa de mortalidade de pequenos negócios, uma vez que muitos empreendedores iniciam as suas atividades sem conhecimentos básicos sobre fluxo de caixa, custos e precificação.

A tecnologia da informação apresenta-se como um vetor fundamental para mitigar estas deficiências, promovendo eficiência operacional e competitividade ([SILVA; BRAGATO, 2025](#)). No entanto, ao procurar ferramentas de auxílio, o microempreendedor depara-se frequentemente com barreiras de acesso a [ERP](#). Conforme apontado por [Coutinho e Souza \(2023\)](#), as barreiras para empresas de pequeno porte incluem não apenas o custo, mas a complexidade excessiva de soluções desenhadas para grandes corporações, que exigem estruturas e conhecimentos que fogem à realidade do [MEI](#).

## 1.1 O problema de pesquisa

Apesar da vasta oferta de software de gestão no mercado, o [MEI](#) enfrenta uma lacuna significativa de ferramentas que traduzam dados operacionais em inteligência de negócio. Estudos recentes, a exemplo de [Silva e Bragato \(2025\)](#), evidenciam que o processo de informatização é determinante para o aumento da eficiência operacional e do faturamento de pequenos negócios. No entanto, muitas soluções comerciais falham em entregar esse valor, focando excessivamente em obrigações fiscais complexas e negligenciando respostas básicas do dia a dia, como “qual o meu lucro real?”.

A ausência de clareza sobre a saúde financeira, somada à complexidade das ferramentas existentes, leva frequentemente a decisões baseadas na intuição e não em fatos. Diante deste cenário, define-se o seguinte problema de pesquisa: **Como um sistema de informação pode oferecer ao [MEI](#) uma visão clara da sua movimentação**

financeira e da lucratividade real dos seus produtos, de forma simplificada e direta?

## 1.2 Objetivos

O objetivo geral deste trabalho é desenvolver e validar um sistema web de gestão financeira e de inventário, focado em fornecer ao [MEI](#) ferramentas visuais para o monitoramento do fluxo de caixa e análise de desempenho de vendas. Para o alcance desse propósito, definem-se os seguintes objetivos específicos:

1. Desenvolver uma aplicação web robusta utilizando a linguagem PHP e o framework Laravel, garantindo a segurança e a integridade dos dados através de uma arquitetura MVC;
2. Implementar funcionalidades de *business intelligence* simplificado, utilizando o processamento de dados transacionais reais para gerar gráficos de movimentação financeira e identificar produtos com maior margem de lucro;
3. Validar a eficácia da ferramenta através de um estudo com potenciais usuários da aplicação, avaliando se a apresentação visual dos dados auxilia efetivamente na tomada de decisão estratégica.

## 1.3 Metodologia

Para alcançar os objetivos propostos, este trabalho adota uma abordagem de pesquisa aplicada de natureza tecnológica, fundamentada na *Design Science Research*. O percurso metodológico estrutura-se nas seguintes etapas:

1. Analisar as principais dificuldades de gestão financeira enfrentadas por microempreendedores, com foco na visibilidade de custos e rentabilidade;
2. Projetar uma arquitetura de software que simplifique a visualização de fluxos de caixa e estoques para usuários sem formação contábil;
3. Desenvolver uma ferramenta computacional que transforme dados operacionais brutos em indicadores de desempenho visual (gráficos de lucro e movimentação);
4. Avaliar a utilidade da solução proposta quanto ao suporte à tomada de decisão e facilidade de uso no cotidiano do [MEI](#).

## 1.4 Organização do trabalho

O restante deste trabalho está estruturado como segue. O Capítulo 2 apresenta a fundamentação teórica, discutindo o panorama do MEI no Brasil, a importância da gestão financeira segundo Almeida e Silva (2024) e os desafios da informatização abordados por Coutinho e Souza (2023). O Capítulo 3 detalha o processo de engenharia de software, descrevendo a arquitetura da aplicação, a modelagem da base de dados e as decisões técnicas tomadas durante a implementação do sistema em Laravel. O Capítulo 4 discute os dados obtidos na validação com os utilizadores, apresentando a análise dos questionários e a discussão sobre a eficácia do sistema proposto. Por fim, o Capítulo 5 apresenta as considerações finais, as limitações identificadas durante o desenvolvimento e sugestões para trabalhos futuros.

## 2 Fundamentação Teórica

Este capítulo apresenta os conceitos e as tecnologias que fundamentam o desenvolvimento de sistemas *web* modernos. A compreensão dessas bases é essencial para garantir que o software atenda a requisitos de escalabilidade, segurança e manutenibilidade, pilares fundamentais da Engenharia de Software.

### 2.1 Padrões de Arquitetura e Organização de Dados

O padrão de arquitetura *Model-View-Controller* (**MVC**) é amplamente utilizado para a organização de aplicações que necessitam de uma clara separação de responsabilidades. Segundo [Otwell \(2025\)](#), essa estrutura divide a aplicação em três camadas lógicas independentes:

- **Model (Modelo):** Gerencia os dados e as regras de negócio. É a camada responsável pela persistência e pela lógica de validação das informações;
- **View (Visão):** Representa a interface com a qual o usuário interage. Sua função é exibir os dados fornecidos pelo modelo de forma visual;
- **Controller (Controlador):** Atua como intermediário, processando as requisições do usuário, acionando o modelo e determinando qual visão deve ser renderizada.

O uso de *frameworks* modernos promove a produtividade ao oferecer ferramentas nativas para mapeamento objeto-relacional (*Object-Relational Mapping* (**ORM**)), garantindo que a interação com bancos de dados ocorra por meio de abstrações de alto nível em vez de consultas SQL brutas ([OTWELL, 2025](#)).

### 2.2 Evolução das Interfaces: Monólitos e SPAs

A evolução das aplicações *web* trouxe diferentes abordagens para o gerenciamento de estado e navegação. As arquiteturas tradicionais, conhecidas como monólitos, processam as rotas e a renderização inteiramente no servidor. Embora robustas, essa abordagem pode resultar em uma experiência de usuário menos fluida devido à necessidade de recarregar a página a cada interação.

Em contrapartida, as *Single Page Application* (**SPA**) delegam a renderização para o navegador do cliente. Isso permite transições instantâneas e uma percepção de desempenho

superior, uma vez que apenas os dados necessários são trafegados via *Application Programming Interface* (API) ([Inertia.js, 2024](#)). O conceito de “Monólito Moderno” surge como uma alternativa para unir a simplicidade de desenvolvimento de um servidor centralizado com a reatividade de uma SPA no *frontend*.

## 2.3 Desenvolvimento Frontend e Tipagem Estática

O desenvolvimento de interfaces modernas baseia-se no paradigma declarativo, onde a interface é descrita em função do seu estado. Bibliotecas baseadas em componentes utilizam o conceito de *Document Object Model* (DOM) para otimizar atualizações visuais, garantindo que apenas os elementos alterados sejam re-renderizados ([Meta Platforms Inc., 2024](#)).

A robustez do código em larga escala é reforçada pela adoção de tipagem estática. O uso de linguagens tipadas permite a detecção de erros em tempo de desenvolvimento, definindo contratos claros para as estruturas de dados que trafegam entre o servidor e a interface ([Microsoft, 2024](#)).

## 2.4 Gestão de Estoque e o Princípio de Pareto

A eficiência na gestão de inventário é um fator crítico para a saúde financeira de microempresas. Uma das metodologias mais eficazes para o controle de estoque é a Curva ABC, fundamentada no Princípio de Pareto. Segundo essa teoria, aproximadamente 80% do faturamento de uma empresa provém de apenas 20% dos itens do catálogo. A classificação mercadológica permite que o gestor identifique quais produtos exigem maior rigor no controle de reposição e negociação com fornecedores ([GS1, 2012](#)).

## 2.5 Análise Preditiva e Tendências

Para o planejamento de longo prazo, a aplicação de métodos estatísticos é fundamental. A Regressão Linear Simples, baseada no método dos Mínimos Quadrados, permite analisar séries históricas de vendas para projetar o faturamento provável de períodos futuros. Ao identificar o coeficiente de tendência da demanda, o sistema oferece ao empreendedor uma ferramenta de planejamento baseada em dados matemáticos, reduzindo a dependência da intuição no processo de decisão estratégica.

## 2.6 Arquiteturas Multi-inquilino e Computação em Nuvem

A transição para o modelo de *Software as a Service* (SaaS) exige que a aplicação suporte a arquitetura multi-inquilino (*multitenancy*), permitindo que uma única instância do software atenda a diversos clientes de forma isolada. Conforme destacam Chong e Carraro (2006), o desafio reside em equilibrar a eficiência do compartilhamento de recursos com a segurança rigorosa dos dados de cada inquilino.

## 2.7 Sistemas de Apoio à Decisão e Inteligência de Negócio

Sistemas de Apoio à Decisão são ferramentas computacionais que auxiliam gestores no processo de escolha estratégica através da análise de dados históricos e projeções (TURBAN et al., 2010). No contexto do BI, a transformação de registros transacionais brutos em indicadores-chave de desempenho permite uma visão clara sobre a saúde financeira da organização (TURBAN et al., 2010).

Conceitos como margem de contribuição, faturamento e ticket médio tornam-se métricas fundamentais que, quando processadas por algoritmos estatísticos e apresentadas de forma visual, reduzem a carga cognitiva do gestor e minimizam decisões baseadas puramente na intuição. Essa abordagem tecnológica é essencial para converter o volume de dados operacionais em conhecimento estratégico voltado à sustentabilidade do negócio (TURBAN et al., 2010).

## 2.8 Qualidade de Software e Verificação Contínua

A garantia de qualidade em Engenharia de Software é obtida por meio de ciclos de validação contínua. O *Test-Driven Development* (TDD) é uma técnica onde os testes são escritos antes da implementação da funcionalidade, servindo como guia para o desenvolvimento e garantindo que o código atenda estritamente aos requisitos definidos (BECK, 2003).

## 2.9 Trabalhos Relacionados e Soluções de Mercado

A literatura acadêmica e o mercado de *software* oferecem diversas abordagens para a gestão de micro e pequenas empresas. No entanto, o processo de informatização do MEI esbarra em limitações específicas. Esta seção analisa trabalhos acadêmicos e soluções comerciais existentes, evidenciando as lacunas que o sistema proposto visa preencher.

### 2.9.1 Perspectiva Acadêmica sobre a Informatização do MEI

Estudos recentes reforçam a importância da adoção de sistemas de informação para a sobrevivência de pequenos negócios. [Silva e Bragato \(2025\)](#) evidenciam que o processo de informatização é determinante para o aumento da eficiência operacional e do faturamento. Contudo, a adoção tecnológica não é trivial. Conforme apontado por [Coutinho e Souza \(2023\)](#), as barreiras para empresas de pequeno porte ao tentarem adotar sistemas ERP incluem não apenas o custo financeiro, mas principalmente a complexidade excessiva de soluções desenhadas para grandes corporações. Estas soluções exigem estruturas e conhecimentos prévios que fogem à realidade diária do MEI.

### 2.9.2 Análise de Soluções de Mercado

Conforme contextualizado no capítulo introdutório, o MEI que busca profissionalizar sua gestão esbarra em limitações ao recorrer às principais opções do mercado:

- **ERPs Comerciais Tradicionais:** Embora robustos, apresentam barreiras financeiras (assinaturas mensais) e operacionais. Como sua arquitetura visa atender desde pequenas até médias empresas com demandas fiscais complexas, acabam entregando uma interface sobrecarregada e subutilizada pelo microempreendedor.
- **Controles Manuais ou Planilhas Eletrônicas:** Costumam ser a alternativa imediata devido à gratuidade. No entanto, exigem alto esforço manual e não oferecem inteligência analítica (BI) automatizada para apoiar decisões rápidas e estratégicas.

### 2.9.3 Posicionamento do Sistema Proposto

Diferente das soluções analisadas, o sistema desenvolvido neste trabalho atua exatamente na lacuna entre a gratuidade (porém com limitação técnica) das planilhas e a complexidade engessada e paga dos ERPs tradicionais. Ao utilizar o *framework* Laravel para uma arquitetura multi-inquilino segura e React para uma interface responsiva baseada em *Client-Side Rendering* (CSR), a solução foca na experiência do usuário de forma acessível.

O diferencial estratégico em relação aos trabalhos correlatos é a democratização de ferramentas de BI de forma gratuita. Em vez de apenas registrar entradas e saídas, o sistema automatiza lógicas complexas de forma transparente para o usuário, como a classificação de inventário pela Curva ABC (Princípio de Pareto) e a análise preditiva de faturamento utilizando Regressão Linear Simples, entregando um valor que o mercado costuma restringir a planos pagos. A tabela ?? sintetiza o comparativo das funcionalidades propostas em relação ao cenário atual.

Tabela 1 – Comparativo entre a solução proposta e as ferramentas existentes

| Funcionalidade / Característica | Sistema Proposto | ERPs Tradicionais   | Planilhas |
|---------------------------------|------------------|---------------------|-----------|
| Acesso / Custo                  | Gratuito         | Pago (Mensalidade)  | Gratuito  |
| Foco exclusivo no MEI           | Sim              | Não                 | Sim       |
| Baixo custo cognitivo           | Sim              | Não                 | Variável  |
| Curva ABC automatizada          | Sim              | Sim (Módulos pagos) | Não       |
| Previsão de demanda             | Sim              | Não                 | Não       |
| Isolamento de dados sistêmico   | Sim              | Sim                 | N/A       |

Fonte: Elaborado pelo autor.

## 2.10 Considerações finais

Neste capítulo, foram explorados os fundamentos teóricos que sustentam o desenvolvimento de aplicações modernas e as bases da gestão financeira corporativa. Adicionalmente, o levantamento das soluções de mercado e trabalhos relacionados evidenciou a necessidade de um sistema acessível e inteligente para o microempreendedor. Esses conceitos estabelecem o embasamento necessário para a compreensão das decisões técnicas e algoritmos implementados na solução, os quais serão detalhados no capítulo a seguir.

## 3 Desenvolvimento e Implementação

Este capítulo detalha a execução técnica do sistema, fundamentada em decisões de engenharia de software voltadas à escalabilidade e à segurança. A seguir, descrevem-se a arquitetura de dados, os fluxos de persistência no *backend* e as escolhas de implementação que sustentam a operação do software.

### 3.1 Modelagem da Base de Dados e Persistência

A arquitetura de dados foi projetada para garantir a integridade referencial e o desempenho em consultas complexas. Para a persistência das informações, adotou-se o Sistema de Gerenciamento de Banco de Dados (SGBD) relacional MySQL. A gestão do esquema foi inteiramente conduzida por meio das *Migrations* do Laravel, o que permitiu o tratamento do banco de dados como código (*Infrastructure as Code*). Essa prática assegura a reprodutibilidade do ambiente e o versionamento de toda a estrutura de tabelas e índices.

O projeto do Diagrama Entidade-Relacionamento (DER) priorizou o atendimento de requisitos não funcionais essenciais, como a rastreabilidade financeira e o isolamento lógico de dados. A estrutura foi normalizada para evitar redundâncias, utilizando chaves estrangeiras com restrições de integridade (*on delete cascade*) para garantir que a exclusão de registros pai não resulte em dados órfãos no sistema. Os detalhes específicos de cada entidade e seus atributos estão documentados no Apêndice A.

#### 3.1.1 Isolamento de Dados (*Multitenancy*)

Considerando a natureza do sistema como um Software como Serviço (SaaS), a segurança e o isolamento dos dados de cada microempreendedor são prioridades fundamentais. Para isso, a arquitetura de isolamento de dados foi centralizada na tabela **users**. Implementou-se uma estratégia de *Multitenancy* lógica, onde todas as entidades operacionais principais — como **produtos**, **clientes**, **vendas** e **categorias** — possuem uma chave estrangeira **user\_id** que atua como a chave de partição.

Para mitigar falhas humanas e garantir que um usuário jamais acesse dados de terceiros, o isolamento foi automatizado no nível da camada de persistência. Utilizou-se o recurso de *Global Scopes* do Eloquent: ao definir um escopo global nas *Models* base, o sistema intercepta todas as consultas (*SELECT*, *UPDATE*, *DELETE*) e anexa automaticamente a cláusula **WHERE user\_id = ?**, referenciando o ID do usuário autenticado na sessão. Essa abordagem de engenharia impede o acesso cruzado indevido a informações sensíveis,

removendo do desenvolvedor a responsabilidade de filtrar manualmente cada consulta e, conseqüentemente, elevando o padrão de segurança da aplicação.

### 3.1.2 Estrutura de Catálogo: Árvore Mercadológica

Diferente de sistemas simplificados que utilizam apenas uma categoria plana, este projeto implementa uma estrutura hierárquica de quatro níveis para a classificação de produtos. Esta organização, fundamentada no padrão de *Global Product Classification* (GPC), permite uma análise granular do inventário e do desempenho de vendas. A estrutura é composta por:

1. **Departamento (Categoria):** Nível macro de segmentação (ex: Vestuário);
2. **Seção (Subcategoria):** Divisão interna do departamento (ex: Moda Masculina);
3. **Grupo:** Especialização da seção (ex: Camisetas);
4. **Subgrupo:** O nível mais granular e ponto de ancoragem do produto (ex: Manga Curta).

No banco de dados, essa hierarquia é representada pelas tabelas `categorias_arvore`, `subcategorias`, `grupos` e `subgrupos`, interligadas por relacionamentos de um-para-muitos. A implementação no *frontend* utiliza carregamento dinâmico: ao selecionar um nível superior, o sistema realiza requisições assíncronas ao `ArvoreController` para filtrar as opções subsequentes, garantindo que o usuário mantenha a consistência durante o cadastro do produto.

### 3.1.3 Rastreabilidade Financeira e Integridade de Estoque

Para garantir a consistência contábil e evitar a perda de contexto histórico, o sistema adota o princípio da imutabilidade para dados transacionais críticos. A lógica de venda é encapsulada em uma **Transação de Banco de Dados** (*Database Transaction*), garantindo que a criação do registro de venda, a baixa no estoque e a geração do registro financeiro ocorram de forma atômica — ou todos os passos são concluídos com sucesso, ou a operação é revertida (*rollback*).

- **Congelamento de Preços:** Na tabela `venda_items`, o campo `preco_unitario` é persistido no ato da compra. Essa técnica de desnormalização controlada é essencial para que alterações futuras no cadastro do produto não corrompam o valor histórico do faturamento total do sistema;

- **Entidade de Finanças:** Foi implementada a tabela `financas`, que atua como um registro consolidado da transação. Nela, são calculados e armazenados o lucro e a despesa exatos daquela operação. Isso blindo o sistema contra flutuações de custo médio, permitindo que o [MEI](#) visualize sua lucratividade real em períodos retroativos;
- **Gestão de Estoque via Livro-Razão:** O sistema abandona a simples atualização de um campo de saldo. Utiliza-se a tabela `movimentacao_estoques` como um *log* imutável. Cada movimentação é processada por uma **Service Class** dedicada (`EstoqueService.php`), que valida a disponibilidade de saldo antes de autorizar a baixa, prevenindo inconsistências de estoque negativo.

### 3.1.4 Auditoria e Padrão Observer

A transparência na gestão do catálogo é assegurada pela tabela `auditoria_produtos`. Em vez de implementar a lógica de log dentro dos controladores (o que causaria redundância de código), utilizou-se o padrão de projeto **Observer** do Laravel.

Através da classe `ProdutoObserver.php`, o sistema monitora eventos do ciclo de vida da *Model Produto*. Quando o método `updated` é disparado, o *Observer* compara automaticamente o estado anterior do objeto (*original data*) com o novo estado (*dirty data*). Caso existam alterações em campos sensíveis, como preço ou nome, o sistema gera automaticamente uma entrada na tabela de auditoria contendo o usuário responsável, os valores antigo e novo, e o carimbo de tempo (*timestamp*), garantindo total rastreabilidade operacional sem poluir a lógica de negócio principal.

## 3.2 Processamento de Dados e Inteligência de Negócio

A inteligência analítica do sistema está concentrada na camada de controle, especificamente no `EstatisticaController.php`. Diferente de sistemas cadastrais que apenas exibem registros, esta implementação processa grandes volumes de dados transacionais para gerar *insights* estratégicos através de técnicas de [BI](#).

### 3.2.1 Otimização de Performance com Query Builder

Em sistemas que possuem painéis de controle, é comum o uso de *Data Warehouse* (DW) e tabelas com dados pré-agregados para acelerar a leitura das informações. No entanto, para este projeto, optou-se por não utilizar modelagens dimensionais ou tabelas agregadas por uma questão de alinhamento com a realidade do negócio, evitando o excesso de complexidade técnica (*over-engineering*).

A decisão de calcular as métricas diretamente na base de dados principal baseou-se em três motivos fundamentais:

- **Volume de Dados:** O volume transacional diário de um **MEI** não é grande o suficiente para justificar o custo e a complexidade de manter um banco de dados analítico separado;
- **Consistência em Tempo Real:** Tabelas agregadas dependem de rotinas de atualização periódicas. Ao calcular as métricas diretamente na base de vendas, o sistema garante que o **MEI** visualize o seu lucro e o giro de estoque atualizados no exato segundo em que uma venda ocorre;
- **Infraestrutura Enxuta:** Para viabilizar um sistema de baixo custo para o micro-empREENDEDOR, é essencial manter o servidor simples, evitando duplicidade de dados e rotinas pesadas de processamento em segundo plano.

Dessa forma, a estratégia de otimização consistiu em delegar os cálculos diretamente ao **SGBD**. A implementação utiliza o *Query Builder* do Laravel em conjunto com instruções `DB::raw`. Através de funções nativas do *Structured Query Language (SQL)* (como **SUM** e **COUNT**), o sistema calcula a margem de contribuição e o lucro líquido em tempo real. Essa abordagem, devidamente apoiada por índices no banco de dados, permite processar milhares de registros e retornar ao servidor apenas os resultados sumarizados, garantindo um *dashboard* rápido e eficiente.

### 3.2.2 Implementação do Algoritmo de Curva ABC

O sistema automatiza a classificação de inventário baseada no Princípio de Pareto. No `EstatisticaController`, a implementação segue uma lógica de processamento em três etapas:

1. **Extração e Ordenação:** O sistema realiza um *join* entre as tabelas `venda_items` e `produtos`, agrupando o faturamento total por item e ordenando a coleção resultante de forma decrescente;
2. **Cálculo de Percentual Acumulado:** Através de um laço de repetição (*loop*), o algoritmo calcula a participação percentual de cada produto no faturamento total e mantém um somatório acumulado;
3. **Categorização Dinâmica:** Com base no somatório acumulado, o código aplica estruturas condicionais para atribuir as classes: **Classe A** (até 80%), **Classe B** (entre 80% e 95%) e **Classe C** (os 5% restantes).

O resultado é encapsulado em um objeto *JavaScript Object Notation* (**JSON**) que alimenta os gráficos de rosca no *frontend*, permitindo que o **MEI** identifique instantaneamente os itens de maior relevância financeira no seu estoque.

### 3.2.3 Algoritmo de Tendência: Regressão Linear Simples

Para a projeção de faturamento, o sistema implementa o método estatístico dos Mínimos Quadrados. O diferencial técnico desta implementação reside no **tratamento e normalização dos dados** antes do cálculo matemático.

O controlador extrai os dados de vendas dos últimos 30 dias. Para garantir a precisão da reta de tendência, o algoritmo executa uma rotina de preenchimento de lacunas: dias em que não houve vendas são identificados e inseridos na série histórica com valor zero. Sem essa normalização, a inclinação da reta seria distorcida, gerando previsões excessivamente otimistas.

Após a normalização, o sistema calcula os coeficientes da equação da reta ( $y = ax+b$ ), projetando o valor esperado para o próximo período. Esse dado é enviado ao *frontend* para ser plotado como uma linha de tendência sobre o gráfico de barras de faturamento diário, fornecendo uma base matemática para o planejamento do empreendedor.

## 3.3 Visualização de Dados e Interface

A camada de apresentação utiliza uma abordagem de **CSR** seletiva para o Painel de Controle (*Dashboard*), visando maximizar a interatividade. A interface foi construída seguindo uma arquitetura baseada em componentes, utilizando o *framework* React para gerenciar o estado da *User Interface* (**UI**) de forma reativa.

### 3.3.1 Renderização de Gráficos e Componentização

Para a tradução visual dos dados analíticos, utilizou-se a biblioteca **Chart.js** encapsulada em componentes reutilizáveis. A renderização ocorre via *HTML5 Canvas*, o que garante alto desempenho mesmo em dispositivos com hardware limitado, comum entre o público **MEI**.

No arquivo `Estatisticas.tsx`, foram implementados gráficos de linha para monitoramento do faturamento temporal e gráficos de rosca (*doughnut*) para a exibição da participação dos produtos na Curva ABC. A integração é feita via *props*, onde o componente pai (React) recebe o **JSON** do *backend* e faz o *parse* dos dados para os eixos do gráfico.

### 3.3.2 Estratégia de Carregamento Assíncrono e Skeleton UI

Diferente das páginas cadastrais simples, o *Dashboard* adota uma estratégia de carregamento desacoplado. Enquanto a navegação entre telas é gerida pelo Inertia.js, a busca dos dados estatísticos é realizada via **Axios** através de um *endpoint* específico de API.

Ao carregar a visualização geral (`VisaoGeral.tsx`), o sistema renderiza imediatamente a estrutura da página (*Layout*) e exibe componentes de *Skeleton* (espaços reservados animados) enquanto a requisição assíncrona é processada. Essa técnica de engenharia de *frontend* melhora a performance percebida, permitindo que o usuário interaja com o menu lateral e outros elementos da interface enquanto os cálculos pesados de lucratividade são finalizados no servidor.

## 3.4 Tecnologias e Ferramentas Utilizadas

A arquitetura do sistema foi estruturada sobre um ecossistema de ferramentas modernas que priorizam a produtividade, a segurança e a manutenibilidade. A integração entre estas tecnologias permite que o sistema para o MEI seja operado de forma leve e escalável.

### 3.4.1 Ecossistema de Backend e Persistência

A camada de servidor foca na robustez do processamento e na proteção da lógica de negócio:

- **PHP 8.2+**<sup>1</sup>: Linguagem de programação principal, utilizada para a implementação das regras de negócio no servidor e manipulação de dados sensíveis;
- **Laravel 11**<sup>2</sup>: *Framework* que provê a estrutura MVC, o ORM Eloquent e ferramentas nativas para gestão de autenticação via **Laravel Sanctum**;
- **MySQL 8.0**<sup>3</sup>: SGBD relacional responsável pela persistência e integridade dos dados, garantindo performance em consultas de agregação;
- **Laravel Scout**<sup>4</sup>: *Driver* utilizado para indexação e busca performática de itens no catálogo de produtos;

<sup>1</sup> Disponível em: <<https://www.php.net/>>

<sup>2</sup> Disponível em: <<https://laravel.com/>>

<sup>3</sup> Disponível em: <<https://www.mysql.com/>>

<sup>4</sup> Disponível em: <<https://laravel.com/docs/scout>>

- **Ziggy**<sup>5</sup>: Biblioteca essencial para integrar o roteamento nomeado do Laravel diretamente no ecossistema React, facilitando a gestão de *Uniform Resource Locators* (URLs).

### 3.4.2 Ecossistema de Frontend e Interface

A interface foi projetada para garantir uma visualização de dados reativa e intuitiva:

- **React 18**<sup>6</sup>: Biblioteca para construção de interfaces reativas baseadas em componentes, gerenciando o estado da UI de forma eficiente;
- **TypeScript**<sup>7</sup>: Superconjunto que adiciona tipagem estática, garantindo que os dados JSON trocados entre servidor e cliente mantenham sua integridade;
- **Inertia.js 2.0**<sup>8</sup>: Tecnologia que permite o comportamento de SPA sem a necessidade de uma API Representational State Transfer (REST) complexa, mantendo o roteamento no servidor;
- **Material UI (MUI)**<sup>9</sup>: Biblioteca de componentes utilizada para garantir consistência visual, acessibilidade e agilidade no desenvolvimento;
- **Chart.js**<sup>10</sup>: Responsável pela renderização visual de gráficos estatísticos e de tendência em Hypertext Markup Language (HTML)5 Canvas;
- **Tailwind CSS**<sup>11</sup>: Framework utilitário utilizado para garantir a responsividade e o design moderno da interface.

### 3.4.3 Ferramentas de Engenharia e Qualidade

Para garantir a integridade do código e a agilidade no ciclo de desenvolvimento:

- **Vite**<sup>12</sup>: Ferramenta de *build* que otimiza o tempo de compilação do *frontend* e o carregamento dos ativos;
- **PHPUnit**<sup>13</sup>: Framework para execução de testes automatizados de integração e unidade, garantindo a estabilidade das funções críticas;

<sup>5</sup> Disponível em: <<https://github.com/tighten/ziggy>>

<sup>6</sup> Disponível em: <<https://react.dev/>>

<sup>7</sup> Disponível em: <<https://www.typescriptlang.org/>>

<sup>8</sup> Disponível em: <<https://inertiajs.com/>>

<sup>9</sup> Disponível em: <<https://mui.com/>>

<sup>10</sup> Disponível em: <<https://www.chartjs.org/>>

<sup>11</sup> Disponível em: <<https://tailwindcss.com/>>

<sup>12</sup> Disponível em: <<https://vitejs.dev/>>

<sup>13</sup> Disponível em: <<https://phpunit.de/>>

- **Faker PHP**<sup>14</sup>: Biblioteca utilizada para gerar dados fictícios realistas, povoando o sistema durante as simulações e testes de estresse;
- **Git e GitHub**<sup>15</sup>: Ferramentas de controle de versão e hospedagem de código para gestão do repositório do projeto.

### 3.5 Considerações finais

Neste capítulo, foram detalhadas as etapas de desenvolvimento e implementação do sistema, com foco nas decisões arquiteturais e na engenharia de software aplicadas para construir uma ferramenta aderente à realidade e escalabilidade do MEI. A modelagem do banco de dados foi estruturada para garantir a integridade dos registros e a segurança da informação, destacando-se a implementação do isolamento lógico de dados (*multitenancy*) e a rigorosa rastreabilidade das movimentações financeiras e de estoque.

Ademais, a arquitetura *backend* fundamentada no ecossistema Laravel demonstrou-se essencial para a proteção das lógicas de negócio e para o processamento eficiente dos algoritmos de inteligência analítica (BI), a exemplo da classificação automatizada da Curva ABC e da projeção de faturamento por regressão linear. Integrada a essa base sólida, a camada de *frontend* construída com React e Inertia.js viabilizou a entrega de uma experiência de uso fluida e interativa, baseada em componentes reativos e estratégias de carregamento assíncrono.

A consolidação de todas essas ferramentas e tecnologias formou a infraestrutura técnica necessária para que o sistema opere de maneira leve, segura e escalável. Com a base de implementação técnica concluída e devidamente documentada, o próximo capítulo apresentará a materialização desses esforços, detalhando as interfaces gráficas concebidas, as validações de robustez via testes automatizados e os resultados da avaliação prática de usabilidade junto ao público-alvo.

---

<sup>14</sup> Disponível em: <<https://fakerphp.org/>>

<sup>15</sup> Disponível em: <<https://github.com/>>

## 4 Resultados e Discussão

Este capítulo apresenta a materialização do sistema desenvolvido, demonstrando como as decisões de engenharia discutidas anteriormente resultaram em uma ferramenta funcional e estratégica para o MEI. A análise estrutura-se em três frentes complementares: (i) a apresentação das interfaces e fluxos operacionais, focada na experiência do utilizador e na inteligência de negócio; (ii) a validação técnica de robustez e desempenho, realizada por meio de testes automatizados de *backend* e auditorias de *frontend*; e (iii) a avaliação qualitativa de usabilidade conduzida com o público-alvo.

### 4.1 Apresentação das Funcionalidades e Interfaces

O sistema web foi concebido para ser eficiente, prático e intuitivo, apresentando uma curva de aprendizado reduzida e suporte a múltiplas resoluções de tela. A seguir, são apresentados os principais módulos do sistema, com destaque para as funcionalidades de inteligência de negócio e organização mercadológica implementadas. A identidade visual do sistema é apresentada na figura 1.

Figura 1 – Logomarca do sistema.



Fonte: Elaborado pelo autor.

É importante ressaltar que, visando a padronização da experiência do usuário e a otimização do desempenho, todas as interfaces de listagem (tabelas) implementam recursos nativos de ordenação e paginação. O usuário pode clicar no cabeçalho de qualquer coluna para reordenar os dados (ascendente ou descendente) e navegar entre páginas de resultados, garantindo que o sistema mantenha tempos de resposta rápidos mesmo com o crescimento do volume de dados armazenados.

### 4.1.1 Página Inicial e Proposta de Valor

A porta de entrada da plataforma é a *Landing Page* (Figura 2), projetada estrategicamente para comunicar a proposta de valor do sistema de forma clara e objetiva. Desenvolvida com componentes visuais modernos, a interface utiliza uma seção de destaque com chamadas para ação que direcionam novos visitantes ao cadastro e usuários recorrentes ao *Dashboard*.

Figura 2 – Tela Inicial: Hero Section e Navegação



Fonte: Elaborado pelo autor.

A arquitetura de informação da página foi estruturada para destacar os quatro pilares do sistema, conforme detalhado na Figura 3: Gestão de Vendas, Controle de Clientes, Estoque Inteligente e Relatórios Detalhados. Esta apresentação prévia das funcionalidades serve para alinhar a expectativa do usuário antes mesmo do primeiro acesso, reduzindo a taxa de rejeição.

Figura 3 – Detalhamento de Funcionalidades e Benefícios



Fonte: Elaborado pelo autor.

#### 4.1.2 Módulo de Autenticação e Registro

O controle de acesso do sistema foi construído sobre uma arquitetura resistente a falhas, garantindo que erros de operação não comprometam a segurança ou a experiência do usuário. A tela de login (Figura 4) implementa uma dupla camada de validação (no navegador e no servidor) que intercepta requisições inválidas antes que elas gerem processamento desnecessário, fornecendo *feedback* visual imediato em caso de erros.

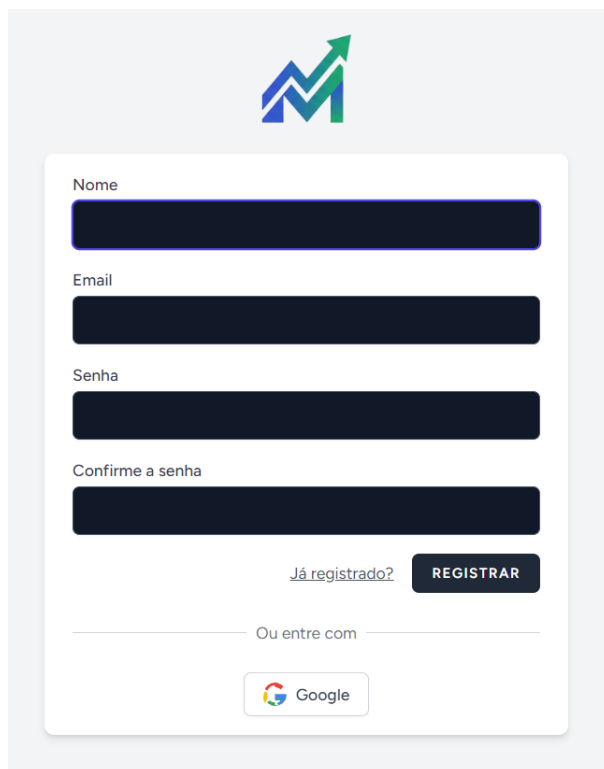
Figura 4 – Interface de Autenticação com Validação de Erros

A interface de autenticação é apresentada em um formulário centralizado sobre um fundo cinza claro. No topo do formulário, há um ícone de uma seta verde apontando para cima, sobreposta a uma linha azul em forma de 'M'. Abaixo do ícone, há dois campos de entrada: 'Email' com o texto 'admin@example.com' e 'Senha' com pontos cinza para ocultar o conteúdo. Abaixo do campo de senha, há uma caixa de seleção desativada com o rótulo 'Lembre de mim'. À direita do campo de senha, há um link azul 'Esqueceu sua senha?' e um botão preto com o texto 'ENTRAR' em branco. Abaixo desses elementos, há uma linha horizontal com o texto 'Ou entre com' no centro. Abaixo da linha, há um botão com o ícone do Google e o texto 'Google'.

Fonte: Elaborado pelo autor.

No fluxo de registro (Figura 5), a resiliência do sistema é evidenciada pela prevenção de inconsistências. O formulário bloqueia a submissão de dados incompletos ou senhas divergentes, assegurando a integridade do cadastro desde o primeiro acesso. Além disso, o tratamento de exceções do *backend* impede a duplicação de e-mails ou a criação de registros corrompidos, garantindo uma base de usuários limpa e segura.

Figura 5 – Formulário de Cadastro com Prevenção de Falhas

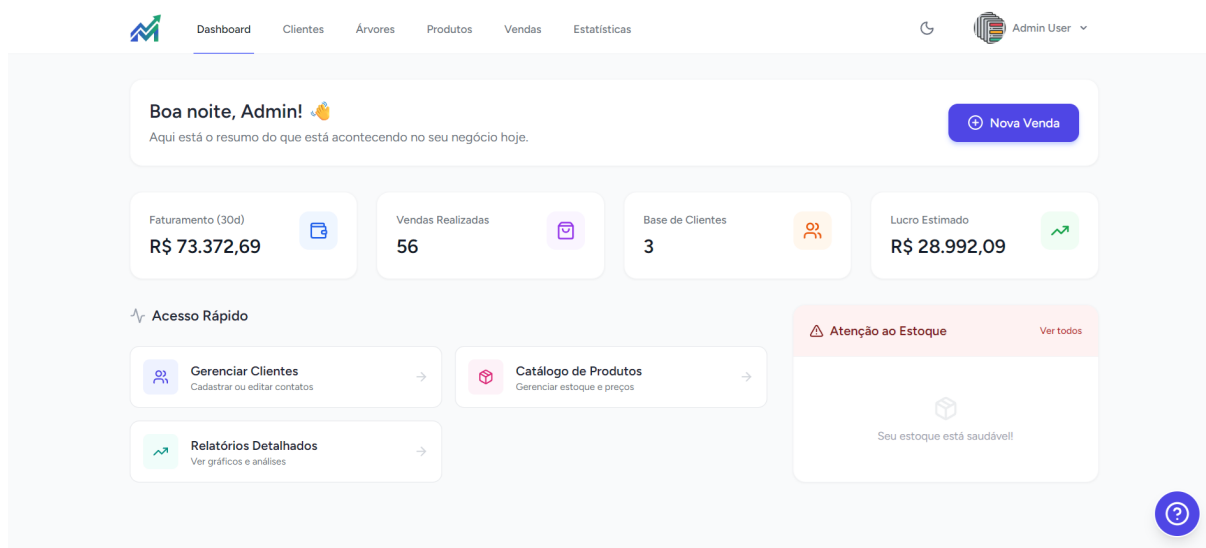
A imagem mostra um formulário de cadastro web. No topo, há um ícone de gráfico de barras com uma seta verde apontando para cima. O formulário contém quatro campos de entrada: 'Nome', 'Email', 'Senha' e 'Confirme a senha'. Abaixo dos campos, há um link 'Já registrado?' e um botão 'REGISTRAR'. Na base, há uma linha divisória com o texto 'Ou entre com' e um botão de login com o ícone do Google.

Fonte: Elaborado pelo autor.

#### 4.1.3 Painel de Controle e Indicadores (Dashboard)

Ao acessar o sistema, o usuário é recebido pelo Painel de Controle (Figura 6). Esta tela consome a API de estatísticas para apresentar o desempenho financeiro em tempo real, incluindo informações do faturamento dos últimos trinta dias, total de vendas realizadas, quantidade de clientes cadastrados, uma estimativa de lucro líquido e alertas sobre produtos com estoque crítico. A interface foi desenhada para entregar um resumo executivo imediato do estado do negócio.

Figura 6 – Painel de Controle com Indicadores de Desempenho

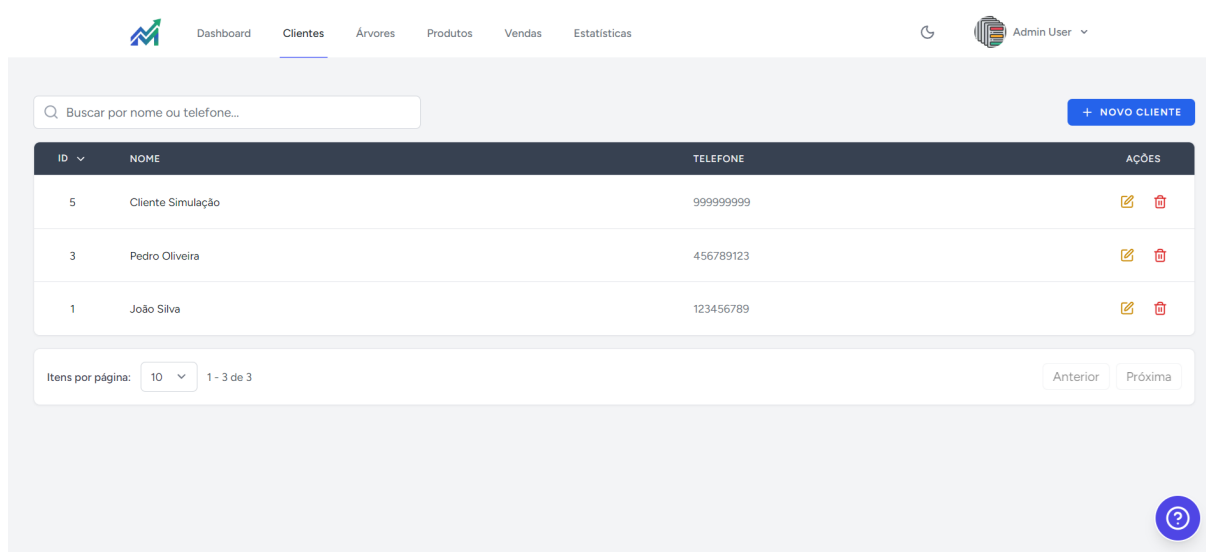


Fonte: Elaborado pelo autor.

#### 4.1.4 Gestão de Clientes

O módulo de gestão de clientes, apresentado na Figura 7, foi desenvolvido priorizando a agilidade no cadastro e consulta de informações. A interface utiliza uma tabela responsiva que exibe dados essenciais e permite a ordenação dinâmica por colunas. Para otimizar a experiência do usuário, as operações de criação e edição são realizadas através de janelas modais, evitando o recarregamento da página e mantendo o contexto de navegação. A comunicação com o servidor é realizada de forma assíncrona, garantindo *feedback* imediato das operações.

Figura 7 – Listagem e Gestão de Clientes



| ID | NOME              | TELEFONE  | AÇÕES                                                                                                                                                                   |
|----|-------------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 5  | Cliente Simulação | 999999999 |   |
| 3  | Pedro Oliveira    | 456789123 |   |
| 1  | João Silva        | 123456789 |   |

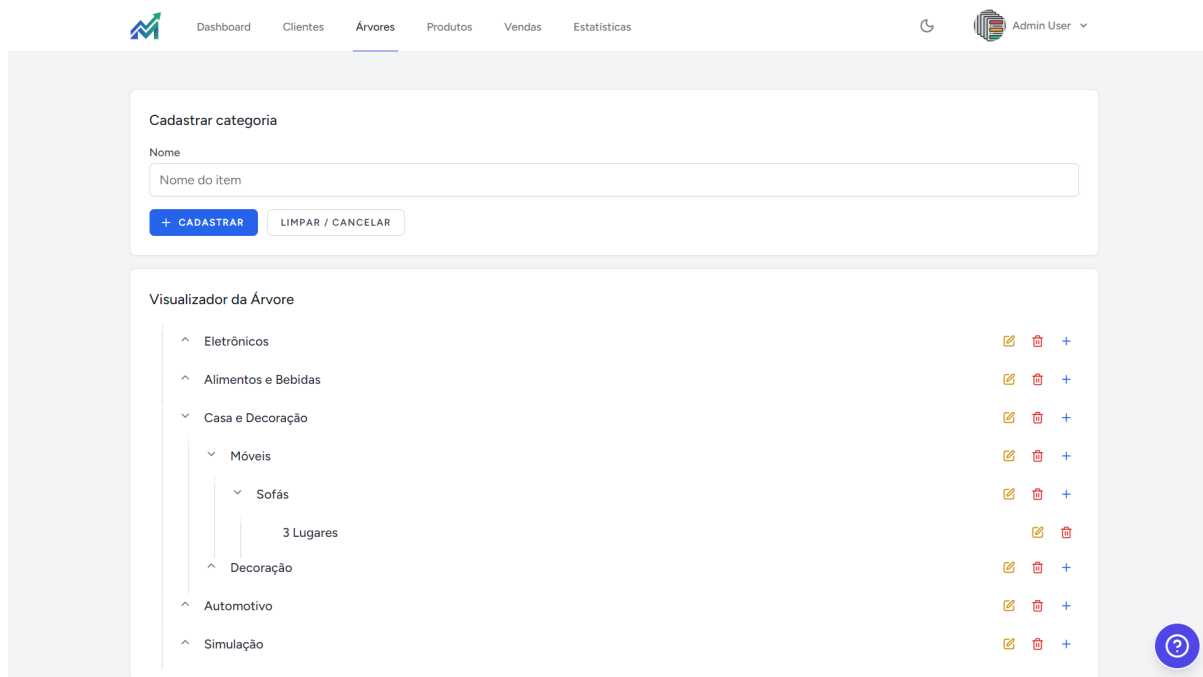
Fonte: Elaborado pelo autor.

#### 4.1.5 Organização Mercadológica (Árvore e Listagem)

A organização mercadológica do catálogo é estruturada por meio de uma hierarquia de quatro níveis — Categoria, Subcategoria, Grupo e Subgrupo — conforme ilustrado na Figura 8. Essa modelagem hierárquica possibilita uma segmentação mais precisa dos produtos, favorecendo a padronização do cadastro, a consistência das informações e a manutenção da organização em um sistema de maior complexidade.

A escolha por uma estrutura mercadológica composta por quatro níveis hierárquicos fundamenta-se em práticas amplamente adotadas em sistemas de gestão empresarial e no varejo em geral, como o padrão GS1 [GPC](#), que adota uma hierarquia de quatro camadas (*Segment*, *Family*, *Class* e *Brick*) para garantir uniformidade global na classificação de produtos ([GS1](#), [2012](#)). Essa abordagem representa um equilíbrio entre o nível de detalhamento necessário para a correta classificação dos produtos e a complexidade operacional do sistema. Estruturas com menor profundidade tendem a gerar agrupamentos excessivamente genéricos, enquanto hierarquias mais profundas aumentam a dificuldade de manutenção, o tempo de cadastro e a probabilidade de inconsistências. Dessa forma, a adoção de quatro camadas mostrou-se adequada para garantir organização, usabilidade e escalabilidade ao sistema proposto.

Figura 8 – Gestão de Catálogo e Estrutura Hierárquica

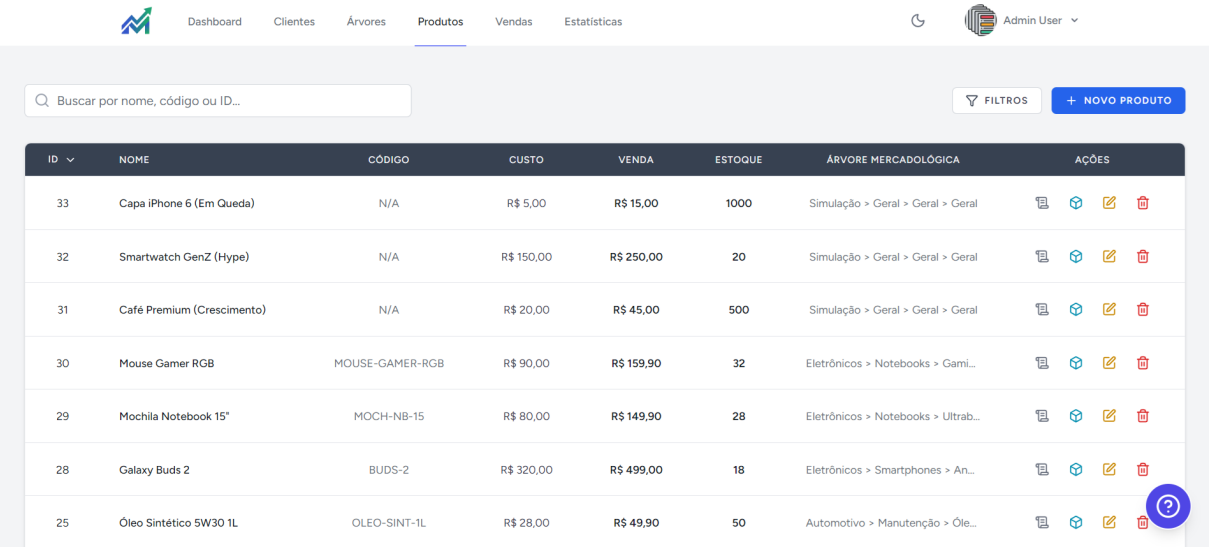


Fonte: Elaborado pelo autor.

#### 4.1.6 Produtos e Controle de Estoque

A listagem de produtos foi projetada para otimizar a recuperação de informações através de um sistema de filtros em cascata fundamentado na árvore mercadológica. A seleção de uma categoria restringe automaticamente as opções dos níveis subsequentes (subcategoria e grupo), garantindo uma segmentação precisa. Para agilizar a rotina operacional, a interface disponibiliza um campo de busca unificado que permite a localização imediata de itens através do nome, código *European Article Number* (EAN) ou *Identifier* (ID) interno, conforme demonstrado na Figura 9.

Figura 9 – Gestão de Produtos com Filtros Avançados



The screenshot displays a web application interface for product management. At the top, there is a navigation bar with tabs: Dashboard, Clientes, Árvores, **Produtos**, Vendas, and Estatísticas. On the right, there is a user profile icon labeled 'Admin User'. Below the navigation bar, there is a search bar with the placeholder text 'Buscar por nome, código ou ID...'. To the right of the search bar are two buttons: 'FILTROS' and '+ NOVO PRODUTO'. The main content area is a table with the following columns: ID, NOME, CÓDIGO, CUSTO, VENDA, ESTOQUE, ÁRVORE MERCADOLÓGICA, and AÇÕES. The table contains seven rows of product data. Each row has a set of four icons in the 'AÇÕES' column: a document icon, a refresh icon, an edit icon, and a delete icon. A blue circular help icon with a question mark is located at the bottom right of the table.

| ID | NOME                       | CÓDIGO          | CUSTO      | VENDA      | ESTOQUE | ÁRVORE MERCADOLÓGICA                | AÇÕES |
|----|----------------------------|-----------------|------------|------------|---------|-------------------------------------|-------|
| 33 | Capa iPhone 6 (Em Queda)   | N/A             | R\$ 5,00   | R\$ 15,00  | 1000    | Simulação > Geral > Geral > Geral   |       |
| 32 | Smartwatch GenZ (Hype)     | N/A             | R\$ 150,00 | R\$ 250,00 | 20      | Simulação > Geral > Geral > Geral   |       |
| 31 | Café Premium (Crescimento) | N/A             | R\$ 20,00  | R\$ 45,00  | 500     | Simulação > Geral > Geral > Geral   |       |
| 30 | Mouse Gamer RGB            | MOUSE-GAMER-RGB | R\$ 90,00  | R\$ 159,90 | 32      | Eletrônicos > Notebooks > Gami...   |       |
| 29 | Mochila Notebook 15"       | MOCH-NB-15      | R\$ 80,00  | R\$ 149,90 | 28      | Eletrônicos > Notebooks > Ultrab... |       |
| 28 | Galaxy Buds 2              | BUDS-2          | R\$ 320,00 | R\$ 499,00 | 18      | Eletrônicos > Smartphones > An...   |       |
| 25 | Óleo Sintético 5W30 1L     | OLEO-SINT-1L    | R\$ 28,00  | R\$ 49,90  | 50      | Automotivo > Manutenção > Óle...    |       |

Fonte: Elaborado pelo autor.

No que tange à manutenção do catálogo, o sistema oferece funcionalidades robustas de cadastro e auditoria. O formulário de criação solicita dados essenciais como classificação mercadológica, custos e estoque inicial. Um diferencial técnico desta etapa é o algoritmo de precificação inteligente: conforme o usuário insere os preços de custo e venda, o sistema calcula a margem de contribuição em tempo real. Caso a margem seja inferior a 30% ou superior a 80%, um alerta visual é exibido sugerindo o preço ideal para garantir a sustentabilidade do negócio. Para a gestão diária, cada item da listagem dispõe de ações rápidas (Figura 10) que permitem realizar ajustes de saldo, visualizar o histórico completo de alterações e editar ou excluir registros sem a necessidade de navegar entre múltiplas telas.

Figura 10 – Ações Rápidas e Movimentação de Estoque

The figure consists of two side-by-side screenshots of a web application interface.

The left screenshot shows a modal window titled "Criar Novo Produto" (Create New Product). It contains the following fields and elements:

- Nome do Produto:** A text input field containing "Notebook Gamer Lenovo Legion Slim 5i i5-13420H 16GB 512GB SSD 83D60003BR".
- Descrição:** A text input field containing "Notebook para jogos, linha lenovo legion".
- Classificação (Árvore Mercadológica):** A series of dropdown menus showing a hierarchy: "Eletrônicos" > "Notebooks" > "Gaming" > "Lenovo Legion".
- Preço de Custo (R\$):** A text input field containing "5000".
- Preço de Venda (R\$):** A text input field containing "7199".
- Margem excelente!** A green notification banner stating "Você está tendo 44% de lucro bruto." (You are getting 44% gross profit).
- Código (EAN/SKU) (Opcional):** A text input field containing "678648378737".
- Estoque Inicial:** A highlighted text input field containing "8".
- Buttons:** "CANCELAR" and "CADASTRAR PRODUTO" (Create Product).

The right screenshot shows two panels related to stock movement for the product "Notebook Gamer Lenovo Legion Slim 5i i5-13420H 16GB 512GB SSD 83D60003BR".

The top panel is titled "Movimentar Estoque" (Move Stock). It shows:

- Estoque Atual:** 8
- Tipo de Movimentação:** A dropdown menu set to "Saída" (Output).
- Quantidade:** A text input field containing "1".
- Descrição/Motivo (Opcional):** A text input field containing "Produto veio danificado" (Product arrived damaged).
- Buttons:** "CANCELAR" and "CONFIRMAR MOVIMENTAÇÃO" (Confirm Movement).

The bottom panel is titled "Histórico do Produto" (Product History). It displays a list of stock movements:

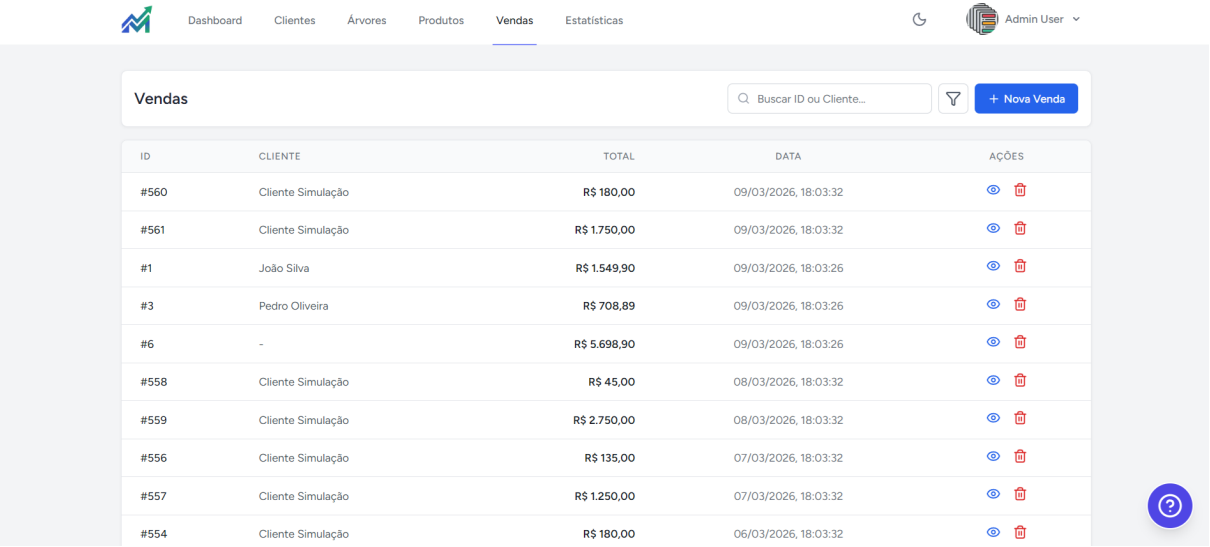
- Evento:** "Estoque" (Stock)
- Ação:** "Saída de 1 un. (Motivo: Produto veio danificado)" (Output of 1 unit (Reason: Product arrived damaged))
- Data e Hora:** "09/03/2026, 18:32:17"
- Evento:** "Estoque" (Stock)
- Ação:** "Entrada de 8 un. (Motivo: Estoque inicial)" (Input of 8 units (Reason: Initial stock))
- Data e Hora:** "09/03/2026, 18:31:19"
- Button:** "FECHAR" (Close)

Fonte: Elaborado pelo autor.

#### 4.1.7 Ponto de Venda (PDV) e Histórico

A gestão de saídas é centralizada na interface de listagem de vendas (Figura 11), que atua como um histórico transacional do negócio. Esta tela oferece ao microempreendedor uma visão cronológica das operações, disponibilizando funcionalidades de auditoria (visualização de detalhes), criação de uma venda e estorno (exclusão), o que permite o cancelamento de lançamentos indevidos de forma prática e segura.

Figura 11 – Histórico de Vendas e Ações Operacionais



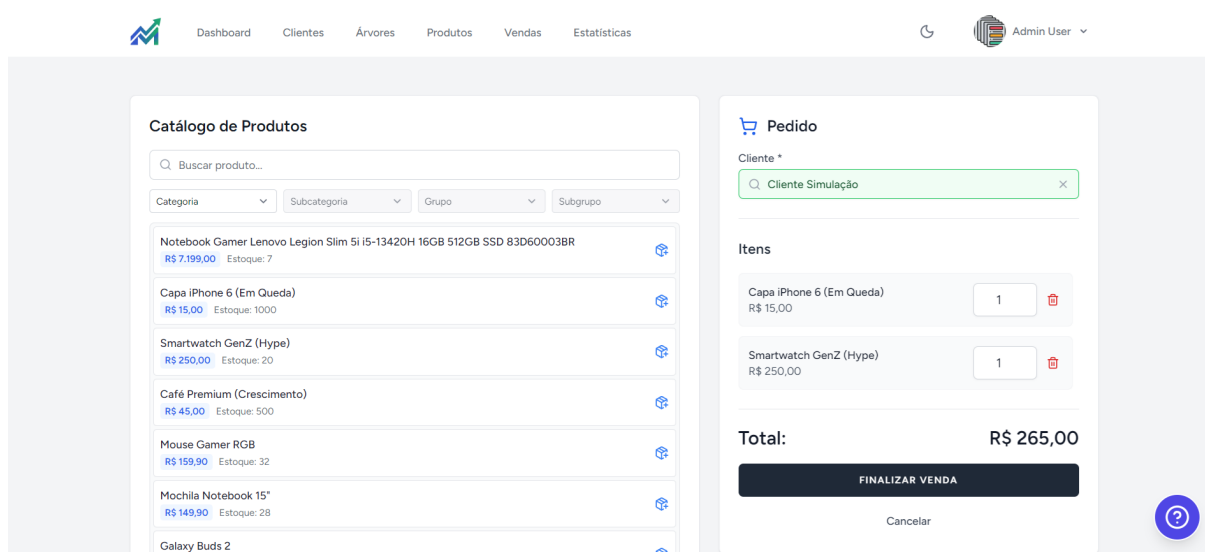
| ID   | CLIENTE           | TOTAL        | DATA                 | AÇÕES                                                                                                                                                                   |
|------|-------------------|--------------|----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| #560 | Cliente Simulação | R\$ 180,00   | 09/03/2026, 18:03:32 |   |
| #561 | Cliente Simulação | R\$ 1.750,00 | 09/03/2026, 18:03:32 |   |
| #1   | João Silva        | R\$ 1.549,90 | 09/03/2026, 18:03:26 |   |
| #3   | Pedro Oliveira    | R\$ 708,89   | 09/03/2026, 18:03:26 |   |
| #6   | -                 | R\$ 5.698,90 | 09/03/2026, 18:03:26 |   |
| #558 | Cliente Simulação | R\$ 45,00    | 08/03/2026, 18:03:32 |   |
| #559 | Cliente Simulação | R\$ 2.750,00 | 08/03/2026, 18:03:32 |   |
| #556 | Cliente Simulação | R\$ 135,00   | 07/03/2026, 18:03:32 |   |
| #557 | Cliente Simulação | R\$ 1.250,00 | 07/03/2026, 18:03:32 |   |
| #554 | Cliente Simulação | R\$ 180,00   | 06/03/2026, 18:03:32 |   |

Fonte: Elaborado pelo autor.

Para a realização de novas operações, o sistema adota um layout estruturado em duas seções lógicas para maximizar a produtividade em desktops, conforme ilustrado na Figura 12. À esquerda, o catálogo visual de produtos facilita a busca e seleção ágil; à direita, o carrinho de compras consolida os itens e calcula os totais financeiros em tempo real.

Um mecanismo crucial de integridade foi implementado na camada de *frontend*: a validação preventiva de estoque. O sistema monitora o saldo disponível de cada produto selecionado e bloqueia instantaneamente a inserção de quantidades excedentes. Essa regra de negócio previne inconsistências no inventário antes mesmo da submissão da venda ao servidor.

Figura 12 – Interface de PDV: Catálogo e Validação de Saldo



Fonte: Elaborado pelo autor.

#### 4.1.8 Módulo de Estatísticas Avançadas e Inteligência de Negócio

Além do painel de controle inicial, o sistema disponibiliza um módulo dedicado à análise de dados profundos, acessível através da página **Estatísticas**. Esta interface foi projetada para centralizar a inteligência do negócio em um único ambiente, segmentando a visualização em duas perspectivas complementares: a **Visão Geral** (análise macroeconômica da empresa) e a **Visão Específica** (análise microeconômica por produto).

#### 4.1.9 Visão Geral: Diagnóstico Macroeconômico

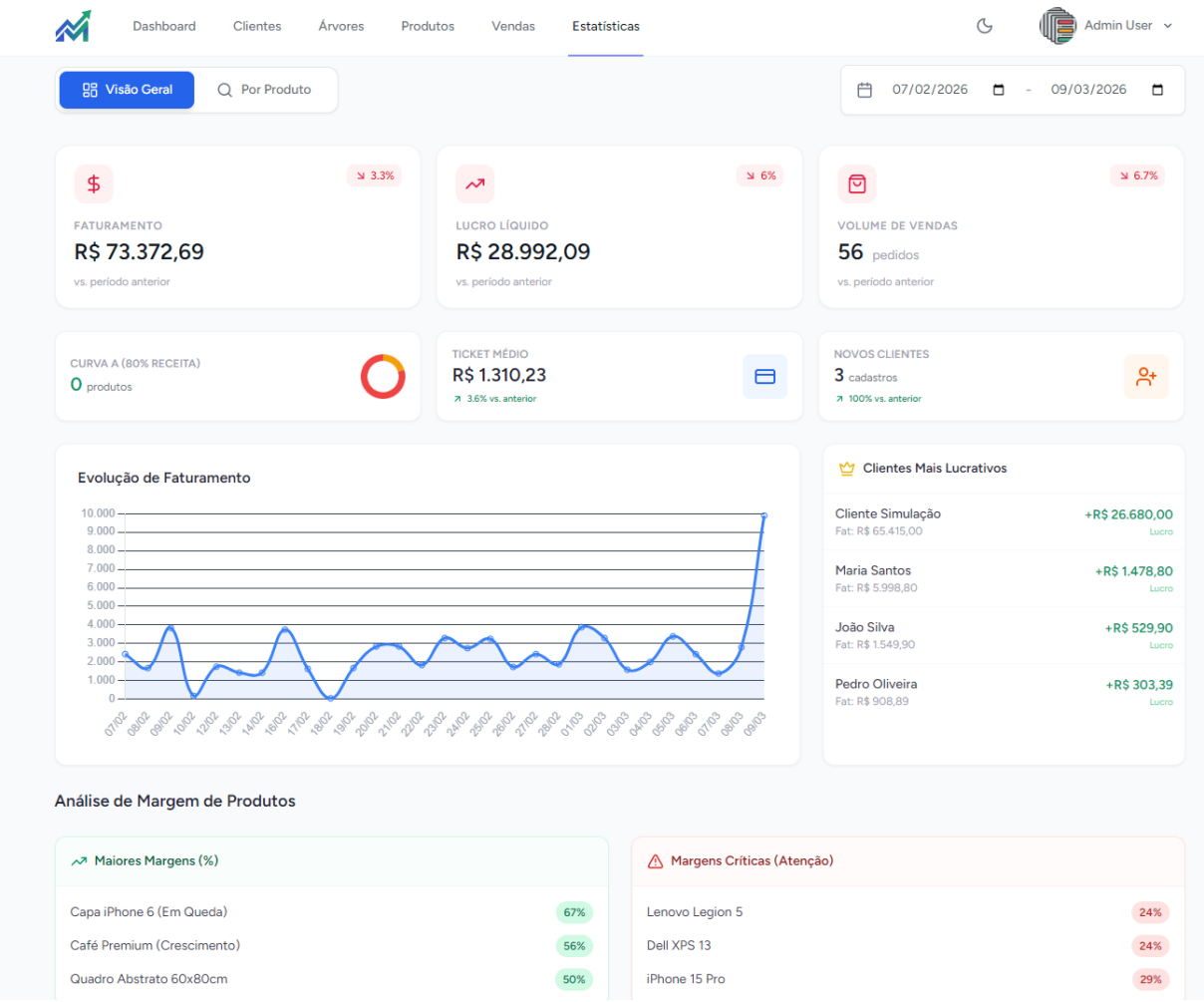
A primeira interface do módulo estatístico fornece um diagnóstico consolidado da saúde financeira da organização. Através do processamento em tempo real de todas as transações de venda registradas no banco de dados, o sistema gera *Key Performance Indicators* (**KPIs**) fundamentais, permitindo que o gestor filtre as informações por intervalos temporais específicos para análises de sazonalidade e fluxo de caixa.

Conforme ilustrado na Figura 13, o painel de Visão Geral sintetiza a inteligência do negócio por meio de métricas que traduzem dados operacionais brutos em conhecimento estratégico para o **MEI**. A estrutura foi desenhada para destacar os seguintes indicadores:

- **Faturamento Total:** Soma bruta de todas as vendas realizadas no período selecionado, servindo como o indicador primário de volume transacional e capacidade de geração de receita operacional;

- **Lucro Líquido:** Resultado da subtração do Custo de Mercadoria Vendida (**CMV**) sobre o faturamento. O sistema realiza este cálculo transação por transação, oferecendo a visão real da margem que permanece efetivamente no caixa da empresa;
- **Volume de Vendas e Ticket Médio:** Enquanto o volume quantifica a atividade comercial, o Ticket Médio revela o gasto médio por cliente. O acompanhamento deste índice é essencial para validar a eficácia de táticas de precificação e estratégias de *upselling*;
- **ABC:** Aplicação automática do Princípio de Pareto para identificar o grupo de produtos responsáveis por 80% do faturamento. Esta informação é vital para prevenir rupturas de estoque nos itens de maior relevância financeira;
- **Taxa de Aquisição de Clientes:** Monitora a expansão da base de consumidores ativos, permitindo ao empreendedor avaliar a atratividade do negócio e o retorno de ações de divulgação local;
- **Análise de Evolução e Rentabilidade:** Gráficos temporais comparam o desempenho atual com períodos anteriores. Adicionalmente, o sistema audita o catálogo para apontar margens de lucro críticas, sugerindo revisões imediatas de preço para garantir a sustentabilidade.

Figura 13 – Visão Geral e Indicadores Macroeconômicos



Fonte: Elaborado pelo autor.

4.1.9.1 Visão Específica (Inteligência de Produto)

A segunda aba do módulo de estatísticas foca na granularidade, aplicando algoritmos de inferência estatística a cada *Stock Keeping Unit* (SKU) individualmente. Essa abordagem permite responder a questões logísticas críticas para o planejamento de compras, como a identificação de produtos em ascensão de demanda ou em processo de queda de desempenho. Tais comportamentos podem ser observados visualmente nas Figuras 14 e 15, que ilustram, respectivamente, cenários de crescimento e declínio no ciclo de vida dos produtos analisados.

4.1.9.2 Métricas Financeiras e Logísticas

- **Faturamento e Lucro por Item:** Isola o desempenho financeiro de um produto específico. Isso permite verificar se itens de alta saída (populares) são, de fato, lucrativos ou se apenas geram volume sem agregar valor real ao negócio.

- **Previsão de Demanda (Forecast):** Utilizando regressão linear sobre o histórico de vendas diárias, o sistema projeta matematicamente o faturamento esperado para o próximo período, auxiliando no planejamento de fluxo de caixa.
- **Volume e Estoque (Giro):** A comparação entre a quantidade vendida e o saldo atual em estoque evidencia a velocidade de giro. Produtos com muito estoque e pouca saída são identificados como capital parado.
- **Previsão de Ruptura (Runway):** Este é um dos indicadores mais sofisticados do sistema. Baseado na velocidade média de vendas, o algoritmo estima em quantos dias o estoque atual chegará a zero. Alertas visuais são disparados quando a previsão é curta ( $< 7$  dias), permitindo a reposição proativa antes que ocorra a perda de vendas.
- **Participação (Market Share Interno):** Uma barra de progresso ilustra a representatividade do produto no faturamento total da empresa, evidenciando o grau de dependência financeira em relação a um único item.

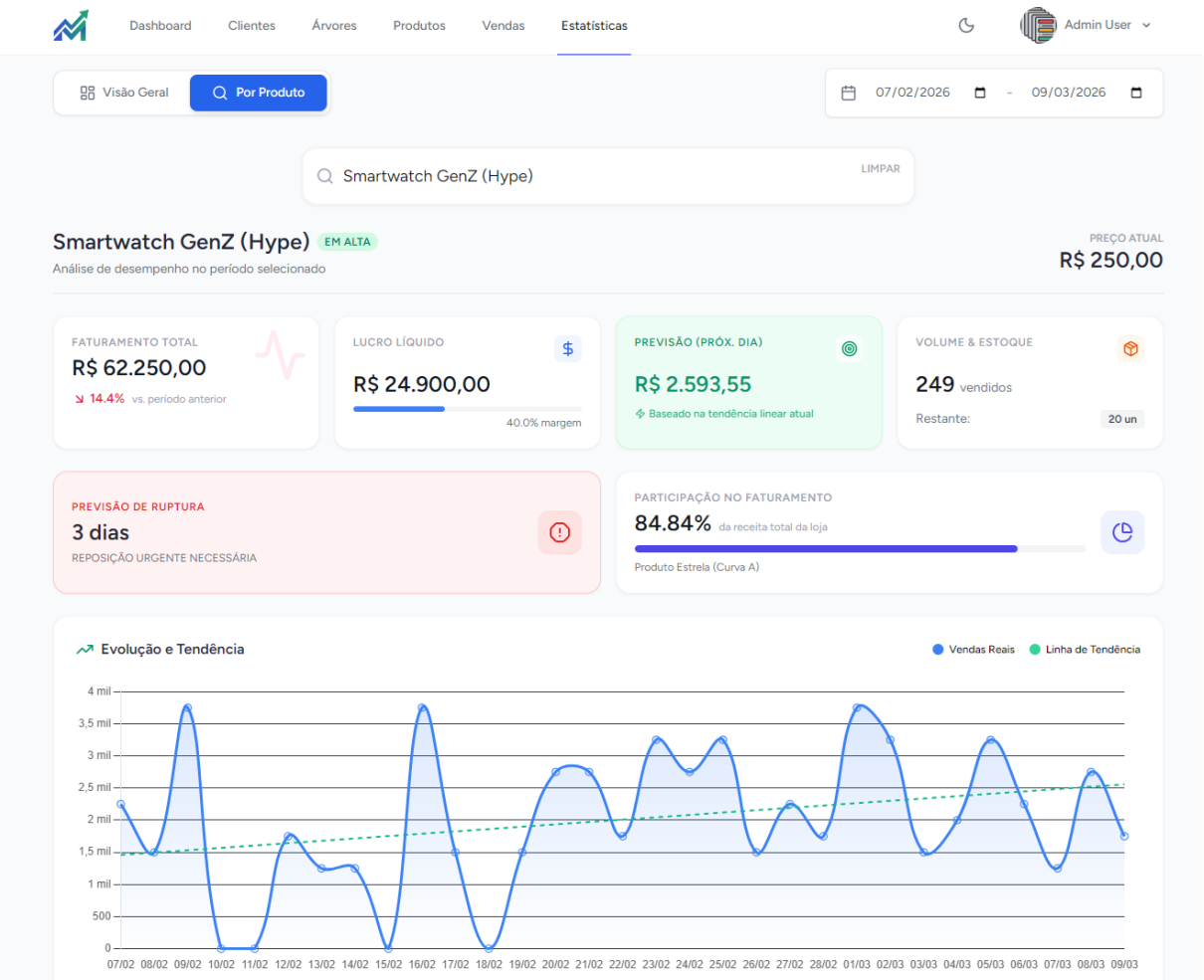
#### 4.1.9.3 Análise de Tendência e Ciclo de Vida

O sistema também avalia o “momento” de mercado do produto através da análise da inclinação da reta de tendência de vendas ( $y = ax + b$ ).

##### 1. Tendência de Alta:

Quando o coeficiente angular é positivo, o sistema sinaliza que o produto está em fase de crescimento de demanda. Conforme ilustrado na Figura 14, indicadores verdes orientam o gestor a garantir estoque para aproveitar o momento favorável.

Figura 14 – Visão Específica (alta)

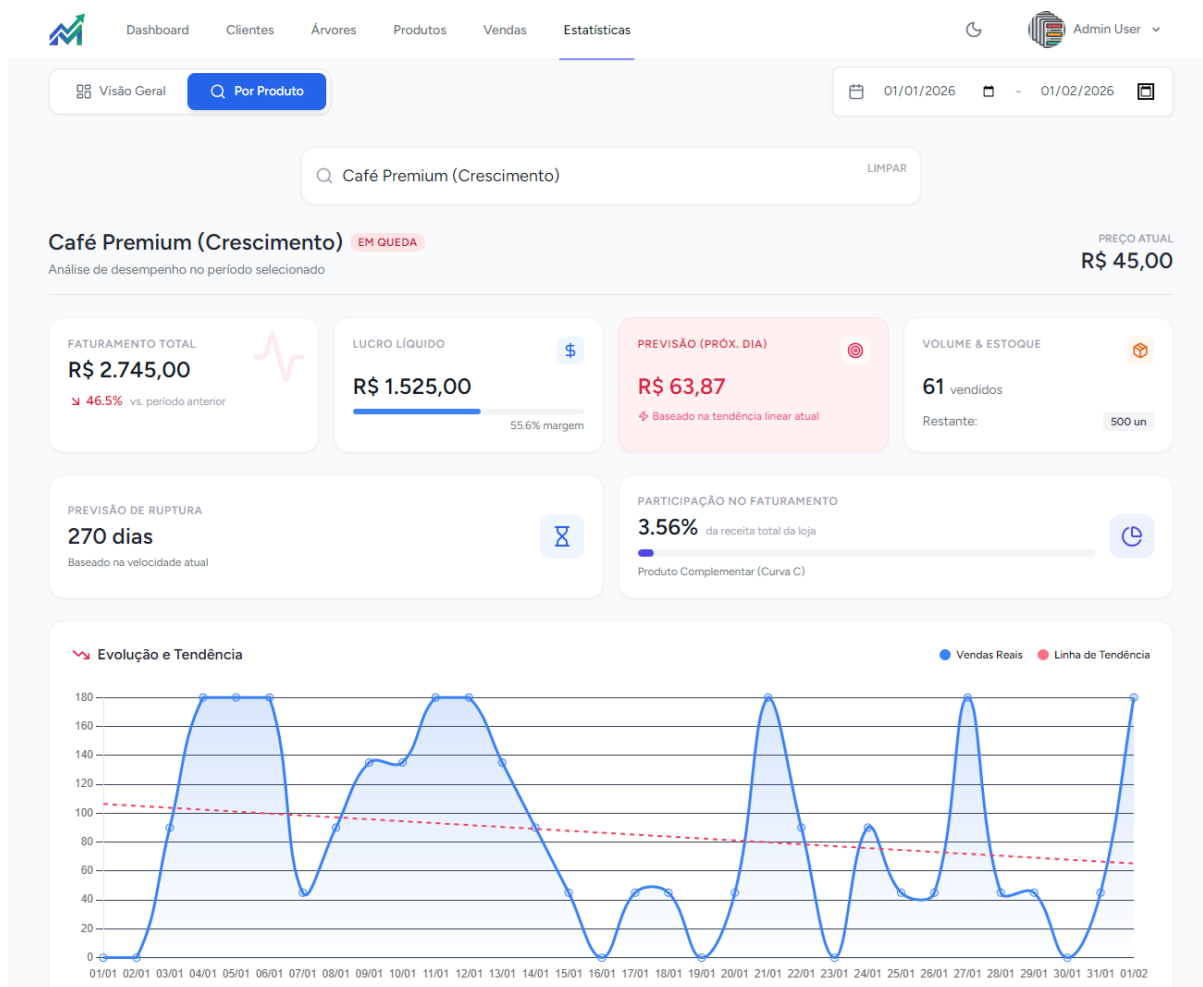


Fonte: Elaborado pelo autor.

2. Tendência de Queda:

Quando o coeficiente é negativo, alertas vermelhos indicam que o produto está perdendo força comercial (Figura 15). Esta informação é crucial para decidir entre realizar promoções de queima de estoque ou descontinuar o item antes que gere prejuízo operacional.

Figura 15 – Visão Específica (queda)



Fonte: Elaborado pelo autor.

### 3. Confiabilidade da Predição (Coeficiente de Determinação):

Para garantir que a tomada de decisão do microempreendedor seja fundamentada, o sistema não apenas projeta a tendência, mas também avalia a precisão matemática dessa projeção através do Coeficiente de Determinação ( $R^2$ ). Este indicador mensura a taxa de ajuste da reta de regressão aos dados reais de vendas, funcionando como uma “taxa de confiabilidade”.

Matematicamente, o  $R^2$  calcula a proporção da variância nas vendas que é explicada pela passagem do tempo. No contexto da interface, essa métrica é traduzida de forma intuitiva: valores próximos a 100% indicam um baixo erro de predição, sinalizando que o comportamento de vendas é consistente e a projeção é altamente confiável. Por outro lado, um índice baixo alerta o gestor de que as vendas daquele item são instáveis e dispersas, indicando que a previsão de demanda possui uma margem de erro maior e deve ser analisada com cautela. Essa abordagem protege o usuário de tomar decisões baseadas em

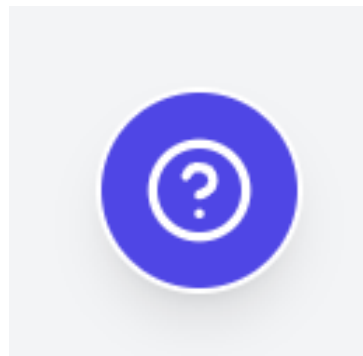
projeções estatisticamente fracas.

#### 4.1.10 Módulo de Onboarding e Ajuda Contextual

Visando mitigar a curva de aprendizado inerente à adoção de sistemas de gestão, foi desenvolvido um módulo de Ajuda Interativa sensível ao contexto. Diferente de manuais estáticos tradicionais, esta funcionalidade detecta dinamicamente a rota ativa do navegador para entregar instruções específicas à tarefa que o usuário está executando no momento.

A interface de acesso segue o padrão de design *Floating Action Button* (FAB). Trata-se de um componente flutuante posicionado estrategicamente no canto inferior direito da tela, identificado pela iconografia universal de ajuda (símbolo “?”), conforme ilustrado na Figura 16. Sua implementação garante que o suporte esteja sempre acessível de forma não intrusiva, sem obstruir a área principal de trabalho.

Figura 16 – Botão de Acesso à Ajuda (FAB)

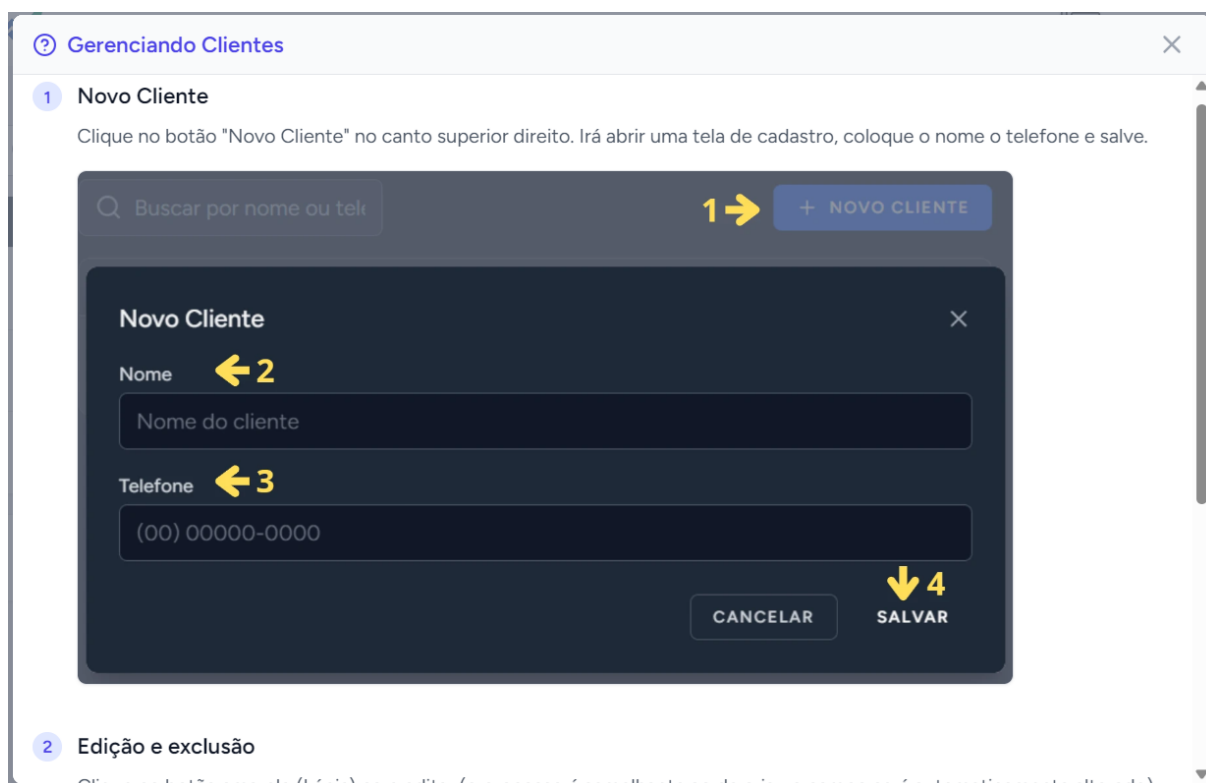


Fonte: Elaborado pelo autor.

Ao acionar o recurso, o sistema consulta o mapa de dados estruturado (`helpContent.ts`) e renderiza uma janela modal sobreposta, preservando o contexto da tela de fundo. A arquitetura técnica utiliza a função `useMemo` do React para otimizar o desempenho, recalculando o conteúdo do tutorial apenas quando ocorre uma mudança efetiva de rota.

O conteúdo é apresentado em formato de guia passo a passo, combinando descrições textuais diretas com capturas de tela ilustrativas, como demonstrado na Figura 17. Esta abordagem visual orienta o microempreendedor na execução correta de processos complexos — desde a estruturação da árvore mercadológica até a finalização de vendas — promovendo a autonomia operacional e reduzindo a dependência de suporte técnico externo.

Figura 17 – Interface do Guia Interativo Passo a Passo



Fonte: Elaborado pelo autor.

#### 4.1.11 Design Responsivo e Mobilidade RWD

Considerando que o microempreendedor moderno frequentemente gerencia suas operações em trânsito, o sistema foi construído seguindo os princípios do RWD e da metodologia *Mobile-First*. A interface utiliza o sistema de *grid* e os *breakpoints* responsivos do *framework* Tailwind CSS para garantir a fluidez da experiência em diferentes resoluções de tela.

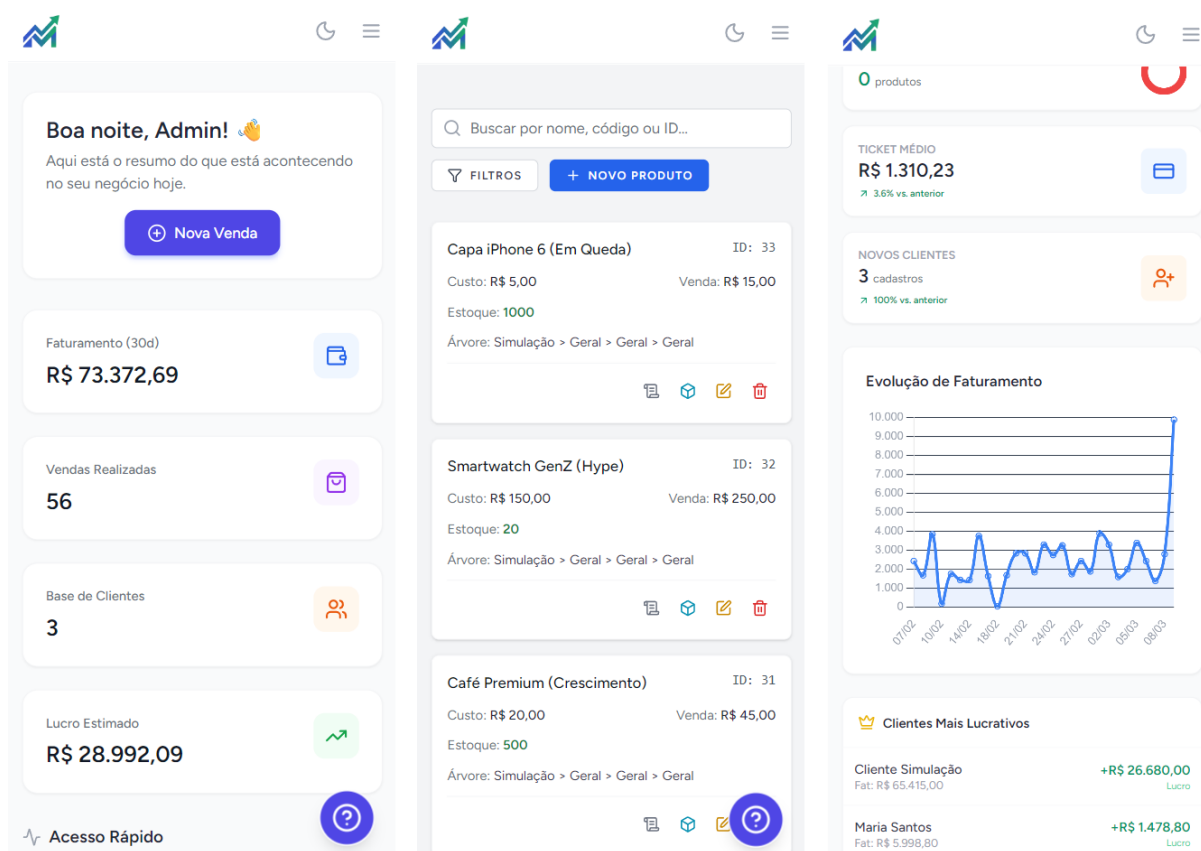
A Figura 18 demonstra as principais estratégias de adaptação implementadas:

1. **Navegação Adaptativa:** Em dispositivos móveis, a barra de navegação horizontal é automaticamente convertida em um menu colapsável, otimizando a área útil da tela para o conteúdo principal.
2. **Transformação de Dados:** Para evitar a rolagem horizontal excessiva — um problema comum de usabilidade em celulares —, as tabelas de listagem de produtos e vendas são substituídas dinamicamente por componentes em formato de *cards* verticais. Esta abordagem preserva a legibilidade das informações sem sacrificar o acesso às ações de edição e exclusão.

3. **Paineis Expansíveis:** Menus complexos, como os filtros da árvore mercadológica, são recolhidos em painéis do tipo *accordion*, permitindo que o usuário foque nos dados e expanda as ferramentas apenas quando necessário.

Essa flexibilidade garante paridade de funcionalidades entre as versões *desktop* e *mobile*, permitindo que o **MEI** realize desde vendas rápidas até análises financeiras complexas diretamente pelo *smartphone*.

Figura 18 – Adaptação da Interface para Dispositivos Móveis



Fonte: Elaborado pelo autor.

## 4.2 Testes Automatizados de Backend

Para garantir a robustez das operações críticas do sistema, como o controle de estoque e o isolamento de dados entre diferentes usuários, foram implementadas suítes de testes automatizados. Abaixo são detalhados os principais cenários validados.

### 4.2.1 Integridade do Estoque

Um dos requisitos mais críticos do sistema é impedir que produtos sejam movimentados sem saldo suficiente. O teste de integridade de estoque (`EstoqueIntegridadeTest`) valida que o sistema lança uma exceção ao tentar realizar uma saída maior que a quantidade disponível.

O código a seguir demonstra a validação dessa regra de negócio diretamente na camada de serviço:

```
// Teste: Não deve permitir saída de estoque insuficiente
public function test_nao_deve_permitir_saida_de_estoque_insuficiente()
{
    $user = User::factory()->create();
    $produto = Produto::factory()->create([
        'user_id' => $user->id,
        'quantidade_estoque' => 5
    ]);

    $service = new EstoqueService();

    // Espera-se que o sistema bloqueie a operação
    $this->expectException(Exception::class);
    $this->expectExceptionMessage('Estoque insuficiente para esta saída.');
```

```
    $service->movimentarEstoque($produto, 'saida', 10);
}
```

### 4.2.2 Fluxo de Vendas

O teste de fluxo de vendas (`VendaFluxoTest`) simula uma transação completa via [API](#). Ele verifica se, ao finalizar uma venda, o sistema realiza três ações simultâneas corretamente: o registro da venda, o cálculo do valor total e a baixa automática no estoque dos produtos envolvidos.

O cenário de teste implementado garante que:

1. O sistema aceita a venda via requisição HTTP;
2. O total da venda é calculado corretamente (soma dos produtos);
3. A quantidade em estoque dos produtos vendidos é decrementada no banco de dados.

### 4.2.3 Segurança e Isolamento de Dados

Considerando que a aplicação pode ser utilizada por múltiplos microempreendedores (MEIs), a segurança dos dados é fundamental. O teste `SegurancaDadosTest` valida o isolamento das informações, garantindo que um usuário autenticado não consiga visualizar, editar ou excluir dados pertencentes a outro usuário.

```
public function test_usuario_nao_pode_ver_produtos_de_outro_usuario()
{
    // Usuário A (Dono do produto) cria um produto
    $userA = User::factory()->create();
    $produtoA = Produto::factory()->create([
        'user_id' => $userA->id,
        'nome' => 'Produto Secreto'
    ]);

    // Usuário B (Intruso) tenta acessar
    $userB = User::factory()->create();
    $this->actingAs($userB);

    // Tenta listar produtos
    $response = $this->getJson('/api/produtos');

    // O JSON retornado NÃO deve conter o produto do Usuário A
    $response->assertJsonMissing(['nome' => 'Produto Secreto']);
}
```

Essa validação garante a privacidade das informações financeiras e de estoque de cada [MEI](#) cadastrado na plataforma.

### 4.2.4 Conclusão dos Testes de Backend

A execução bem-sucedida da suíte de testes automatizados valida a robustez e a confiabilidade das regras de negócio implementadas no *backend*. A cobertura de cenários críticos — abrangendo desde a integridade das transações de estoque até o isolamento rigoroso dos dados entre diferentes usuários — assegura que o sistema opera de maneira consistente e segura.

Como evidenciado na Figura 19, todos os testes foram concluídos com êxito, confirmando que a lógica da aplicação atende plenamente aos requisitos funcionais e de segurança estabelecidos. Essa validação estrutural do servidor fornece a base necessária

para garantir uma experiência de usuário estável, permitindo que a avaliação do sistema avance para os aspectos de desempenho e interface, conforme abordado na seção a seguir.

Figura 19 – Relatório de execução dos testes automatizados

|                                         |                                                                            |       |
|-----------------------------------------|----------------------------------------------------------------------------|-------|
| <b>PASS</b>                             | Tests\Feature\Auth\PasswordUpdateTest                                      |       |
| ✓                                       | password can be updated                                                    | 0.06s |
| ✓                                       | correct password must be provided to update password                       | 0.06s |
| <b>PASS</b>                             | Tests\Feature\Auth\RegistrationTest                                        |       |
| ✓                                       | registration screen can be rendered                                        | 0.03s |
| ✓                                       | new users can register                                                     | 0.12s |
| <b>PASS</b>                             | Tests\Feature\EstoqueIntegridadeTest                                       |       |
| ✓                                       | nao deve permitir saída de estoque insuficiente pelo service               | 0.47s |
| ✓                                       | deve atualizar estoque corretamente apos entrada                           | 0.24s |
| <b>PASS</b>                             | Tests\Feature\ExampleTest                                                  |       |
| ✓                                       | the application returns a successful response                              | 0.07s |
| <b>PASS</b>                             | Tests\Feature\ProfileTest                                                  |       |
| ✓                                       | profile page is displayed                                                  | 0.06s |
| ✓                                       | profile information can be updated                                         | 0.16s |
| ✓                                       | email verification status is unchanged when the email address is unchanged | 0.14s |
| ✓                                       | user can delete their account                                              | 0.03s |
| ✓                                       | correct password must be provided to delete account                        | 0.03s |
| <b>PASS</b>                             | Tests\Feature\SegurancaDadosTest                                           |       |
| ✓                                       | usuario nao pode ver produtos de outro usuario                             | 0.23s |
| ✓                                       | usuario nao pode excluir venda de outro usuario                            | 0.08s |
| <b>PASS</b>                             | Tests\Feature\VendaFluxoTest                                               |       |
| ✓                                       | deve realizar venda baixar estoque e calcular total                        | 0.17s |
| ✓                                       | nao deve vender sem estoque via api                                        | 0.04s |
| ✓                                       | exclusao de venda deve estornar estoque                                    | 0.05s |
| Tests: <b>34 passed</b> (79 assertions) |                                                                            |       |
| Duration: 12.33s                        |                                                                            |       |

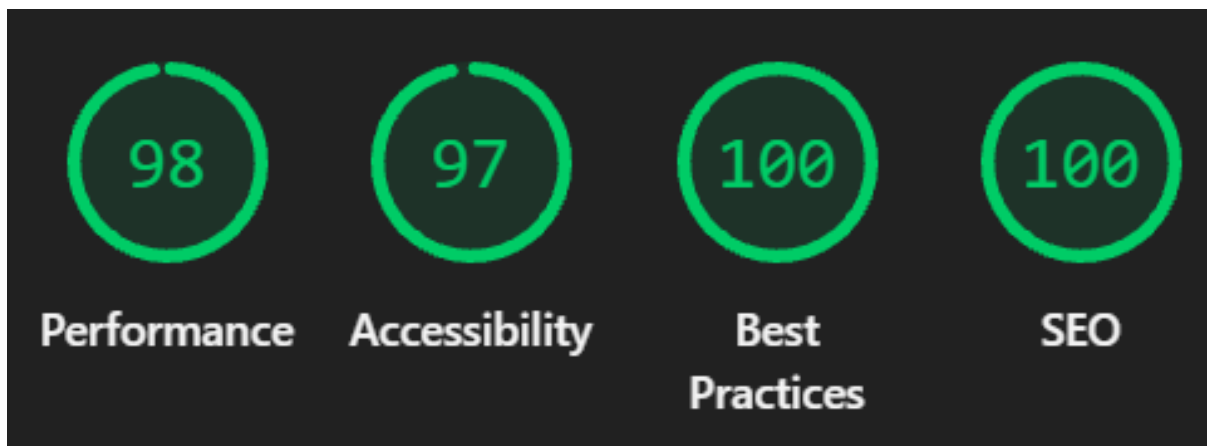
Fonte: Elaborado pelo autor.

### 4.3 Análise de Desempenho e Qualidade (Frontend)

A interface do sistema foi submetida à auditoria do *Google Lighthouse*, uma ferramenta automatizada de código aberto para melhorar a qualidade de páginas *web*. A auditoria avalia métricas de desempenho (*Performance*), acessibilidade (*Accessibility*), boas práticas (*Best Practices*) e Otimização para Motores de Busca (*Search Engine Optimization* ([SEO](#))).

Os resultados obtidos demonstram a eficácia da escolha da *stack* tecnológica e das otimizações implementadas. Como pode ser observado na Figura 20, o sistema obteve pontuação máxima (100) em *Best Practices* e [SEO](#), além de resultados excelentes em *Performance* (98) e *Accessibility* (97).

Figura 20 – Resultado da auditoria Google Lighthouse



Fonte: Elaborado pelo autor com base na ferramenta Google Lighthouse (2025).

O alto desempenho é atribuído ao uso de carregamento assíncrono para componentes pesados e à renderização eficiente do [DOM](#) do React. A acessibilidade foi garantida através do uso de tags semânticas HTML5 e atributos *Accessible Rich Internet Applications* ([ARIA](#)) adequados nos componentes de interface.

#### 4.4 Avaliação da Experiência do Usuário (UX)

A avaliação com usuários finais é o componente qualitativo que valida a eficácia das funcionalidades desenvolvidas. O sistema foi submetido a um teste de usabilidade com uma amostra de 10 participantes, utilizando um formulário estruturado. A pesquisa foi dividida em quatro sessões estratégicas, fundamentais para garantir que a solução atenda aos requisitos de simplicidade e eficiência propostos para o [MEI](#):

- **Sessão 1 - Perfil do Usuário:** Fundamental para identificar a maturidade digital e a experiência de negócio do respondente. O motivo desta sessão é validar se o sistema é resiliente a diferentes níveis de conhecimento técnico e teórico.
- **Sessão 2 - Interface e Navegação:** O objetivo é medir o conforto visual e a clareza da arquitetura da informação. Validar o menu e o "Modo Escuro" é crucial para reduzir o esforço cognitivo do usuário durante a operação.
- **Sessão 3 - Funcionalidades e Regras de Negócio:** Esta sessão foca no "core" do sistema. Busca-se entender se os fluxos complexos (*Árvore Mercadológica* e [PDV](#)) foram simplificados o suficiente para serem operados sem treinamento prévio.

- **Sessão 4 -SUS:** Baseada na escala *System Usability Scale*, esta sessão quantifica a confiança e a probabilidade de recomendação da ferramenta, fornecendo uma métrica de sucesso final.

#### 4.4.1 Resultados e Análise do Feedback

Nesta subseção, detalham-se os dados obtidos. A análise segue o padrão de apresentar o contexto dos resultados seguido pelas evidências visuais coletadas.

A análise do perfil inicial revelou uma amostra equilibrada. Observou-se que mesmo usuários com familiaridade tecnológica média ou baixa conseguiram navegar pelo sistema sem auxílio externo. Quanto à experiência de gestão, o grupo contou tanto com empreendedores experientes quanto com leigos, o que permitiu testar a eficácia dos conceitos didáticos da interface.

Figura 21 – Nível de familiaridade tecnológica

Qual seu nível de familiaridade com uso de sistemas web/computador?

10 respostas

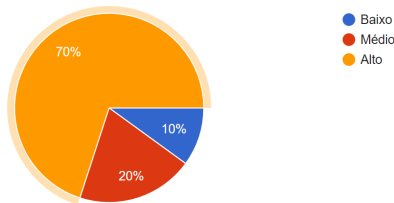
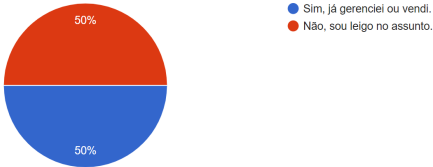


Figura 22 – Experiência prévia em gestão

Você tem experiência com gestão de negócios ou vendas?

10 respostas



Fonte: Elaborado pelo autor (2025).

Sobre os aspectos visuais e de navegação, a recepção foi unânime. O menu lateral foi classificado como "Muito Intuitivo", e a implementação do Modo Escuro recebeu *feedback* positivo em termos de legibilidade e conforto, reforçando a escolha da paleta de cores e tipografia da plataforma.

Figura 23 – Atratividade visual da interface

O visual do sistema (cores, fontes, disposição) é agradável?

10 respostas

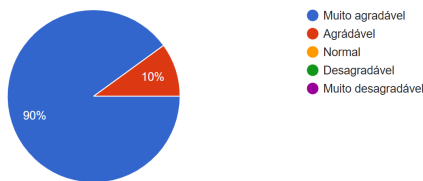
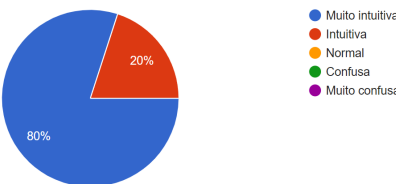


Figura 24 – Clareza do menu principal

A navegação pelo menu principal (Dashboard, Clientes, Produtos, etc.) foi fácil de entender?

10 respostas



Fonte: Elaborado pelo autor (2025).

Figura 25 – Feedback sobre o Modo Escuro

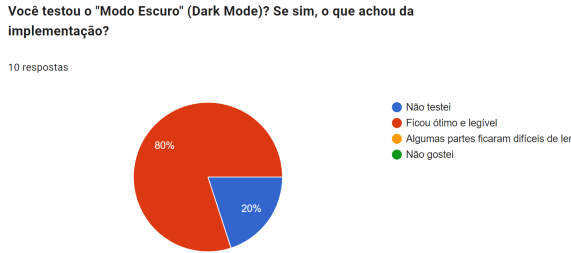
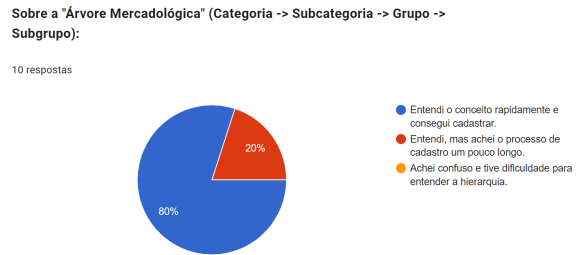


Figura 26 – Compreensão da Árvore Mercadológica



Fonte: Elaborado pelo autor (2025).

No que tange às operações de estoque e vendas, os dados mostram que o fluxo de PDV (Frente de Caixa) é um dos pontos mais fortes da aplicação, sendo descrito como "Muito Rápido". A estrutura de árvore mercadológica, apesar de ser um conceito hierárquico, foi assimilada rapidamente através das categorias e subcategorias.

Figura 27 – Facilidade de cadastro de produtos

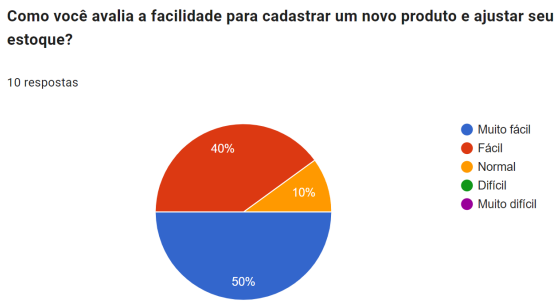


Figura 28 – Fluidez do fluxo de vendas



Fonte: Elaborado pelo autor (2025).

Os resultados da escala de usabilidade (Figuras 31 a 35) consolidam a validação do sistema. Destaca-se que a maioria dos usuários "concorda totalmente" que usaria o sistema com frequência e sente-se confiante ao operá-lo, o que indica uma baixa taxa de erro induzido pelo *design*.

Figura 29 – Utilidade do sistema de ajuda

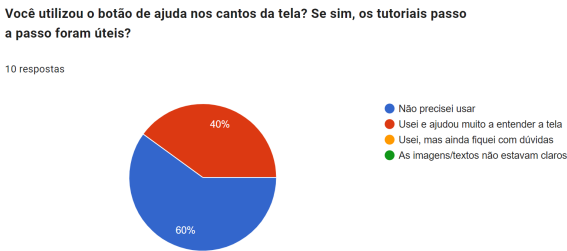
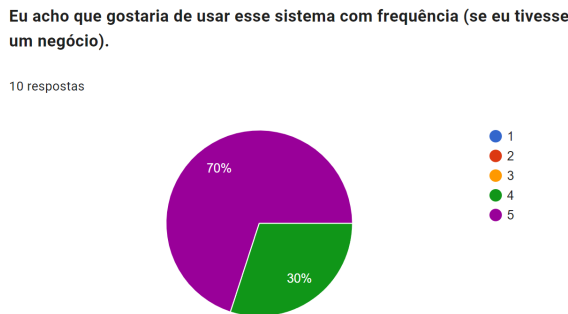


Figura 30 – Desejo de uso frequente



Fonte: Elaborado pelo autor (2025).

Figura 31 – Percepção de complexidade

Eu achei o sistema desnecessariamente complexo.

10 respostas

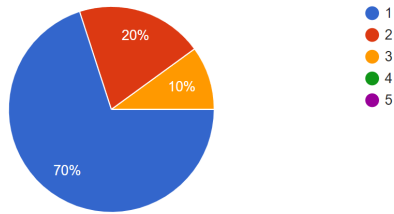
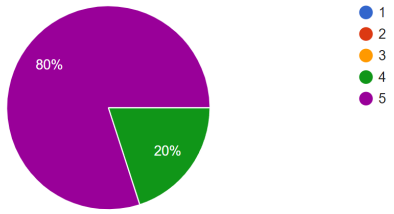


Figura 32 – Facilidade de uso geral

Eu achei o sistema fácil de usar.

10 respostas



Fonte: Elaborado pelo autor (2025).

Figura 33 – Velocidade de aprendizado

Eu acho que a maioria das pessoas aprenderia a usar esse sistema muito rapidamente.

10 respostas

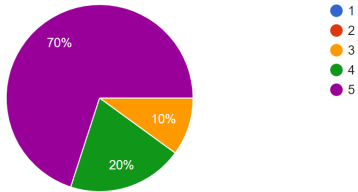
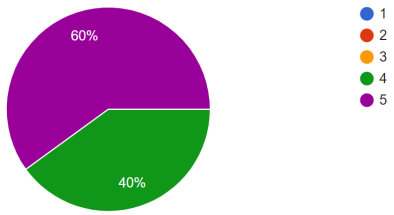


Figura 34 – Nível de confiança no uso

Eu me senti muito confiante usando o sistema.

10 respostas



Fonte: Elaborado pelo autor (2025).

Ao final, os participantes puderam expressar suas opiniões em campos abertos. O módulo de estatísticas avançadas foi citado repetidamente como a funcionalidade de maior valor. A ausência de relatos de *bugs* impeditivos valida a robustez do código e a eficácia das validações de estoque implementadas.

Figura 35 – Funcionalidades favoritas

Qual foi a funcionalidade que você mais gostou ou achou mais interessante?

9 respostas

|                                                                                                                                                             |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Gostei muito do UX design                                                                                                                                   |
| Estoque                                                                                                                                                     |
| INTERESSANTE PARA COMERCIANTE                                                                                                                               |
| A opção "dark mode"                                                                                                                                         |
| Estatística                                                                                                                                                 |
| O Dashboard de estatística. Simplifica e agiliza os cálculos necessários, facilitando a vida do usuário.                                                    |
| Árvore Mercadológica                                                                                                                                        |
| A organização do site                                                                                                                                       |
| O relatório detalhado dos produtos, exibindo as vendas, lucros, probabilidade de tempo para zerar o estoque, dentre outras funcionalidades deste relatório. |

Figura 36 – Relatos de erros ou bugs

Você encontrou alguma dificuldade específica ou "bug" durante o teste? Se sim, descreva.

8 respostas

|                                                                                        |
|----------------------------------------------------------------------------------------|
| Não                                                                                    |
| AINDA NÃO                                                                              |
| Não achei nenhum big perceptível que prejudicasse o usuário na interação com o sistema |
| Não encontrei , perfeito                                                               |
| Não                                                                                    |
| Não, tudo funcionou perfeitamente                                                      |
| Não encontrei dificuldade ou bug.                                                      |

Fonte: Elaborado pelo autor (2025).

Figura 37 – Sugestões de melhorias

|                                                          |                                                                                                                                                                                                                                                                                                    |
|----------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Se você pudesse mudar uma coisa no sistema, o que seria? |                                                                                                                                                                                                                                                                                                    |
| 8 respostas                                              |                                                                                                                                                                                                                                                                                                    |
|                                                          | Nada, tá um mel                                                                                                                                                                                                                                                                                    |
|                                                          | BUSCAR ENTENDER QUAIS IMPOSTOS INSIDE NA ENTRADA E NA VENDA DO PRODUTO E INCLUIR NO SISTEMA PARA FAZER PREÇO                                                                                                                                                                                       |
|                                                          | Dashboard poderia ser mais completo                                                                                                                                                                                                                                                                |
|                                                          | Cadastro de produtos                                                                                                                                                                                                                                                                               |
|                                                          | Não pensei em nenhuma alteração necessária.                                                                                                                                                                                                                                                        |
|                                                          | Nada                                                                                                                                                                                                                                                                                               |
|                                                          | Nada                                                                                                                                                                                                                                                                                               |
|                                                          | No momento, só faria melhorias gerais para deixar o sistema com mais funcionalidades sem fugir muito da proposta para não ficar muito complexo. Isso também depende da demanda de cada empresa (negócio) que implementaria este sistema e precisaria acrescentar alguma funcionalidade específica. |

Fonte: Elaborado pelo autor (2025).

### 4.5 Análise Crítica e Limitações do Projeto

A análise crítica dos resultados permite confrontar as premissas iniciais do projeto com a percepção prática dos usuários, consolidando o aprendizado obtido durante a fase de validação. Apesar da elevada aceitação geral, identificaram-se limitações relevantes, sobretudo a ausência de funcionalidades fiscais (como cálculo de Imposto sobre Circulação de Mercadorias e Serviços (ICMS) e gestão tributária). Tal exclusão foi deliberada, motivada pela elevada complexidade e pela constante mutabilidade da legislação tributária brasileira, que demandariam integrações avançadas e configurações personalizadas. Essa escolha preserva a essência do sistema: uma ferramenta simples e ágil de apoio à decisão financeira interna do MEI, sem pretender substituir um software contábil completo.

No que tange à usabilidade, o sistema foi considerado intuitivo pela maioria, mas o cadastro inicial estruturado (árvore mercadológica) gerou uma percepção de complexidade no primeiro contato devido ao tempo demandado para a configuração do catálogo. Como perspectivas de melhoria, destacam-se: (i) a implementação de um guia interativo assistido (*walkthrough*) no primeiro acesso, visando reduzir a carga cognitiva inicial; e (ii) a adoção de sugestões preditivas de categorias e grupos com base no histórico de cadastros anteriores, acelerando o processo operacional. O desafio das próximas versões será evoluir funcionalmente sem comprometer essas características fundamentais que garantem a autonomia do usuário.

## 4.6 Considerações finais

Este capítulo apresentou a materialização do sistema e sua respectiva validação sob as perspectivas técnica e prática. A execução bem-sucedida dos testes de *backend* e a performance otimizada do *frontend* validaram as decisões de engenharia descritas no Capítulo 3. Complementarmente, a avaliação com o público-alvo ratificou a utilidade da solução para o [MEI](#), destacando o módulo de inteligência de negócio como o principal diferencial competitivo.

Apesar das limitações identificadas, como a ausência de módulos fiscais, os resultados confirmam que o sistema cumpre seu objetivo geral de simplificar a gestão financeira. Com esta base consolidada, o próximo capítulo apresenta as conclusões finais deste trabalho e as propostas para evoluções futuras da plataforma.

## 5 Conclusão

Este trabalho apresentou o desenvolvimento e a validação de um sistema web de gestão financeira e de inventário projetado especificamente para as necessidades do [MEI](#). A pesquisa foi motivada pelo desafio de transformar dados operacionais brutos em inteligência de negócio, visando mitigar a elevada taxa de mortalidade de pequenos empreendimentos decorrente da falta de visibilidade sobre custos e receitas.

### 5.1 Síntese do Trabalho e Alcance dos Objetivos

O objetivo geral de construir uma ferramenta robusta e simplificada foi atingido através da integração de tecnologias modernas de engenharia de *software*. O uso do *framework* Laravel no *backend* assegurou a integridade dos dados e o isolamento lógico entre inquilinos (*multitenancy*), enquanto a interface reativa em React e [MUI](#) proporcionou uma experiência de utilizador fluida e responsiva.

Com base na validação realizada, conclui-se que os objetivos específicos foram cumpridos:

- **Eficiência Técnica:** A auditoria Google Lighthouse confirmou a qualidade do sistema, atingindo pontuações de excelência em *Performance* (98) e *Acessibilidade* (97), além de nota máxima em boas práticas e [SEO](#).
- **Robustez de Regras:** Os testes automatizados validaram cenários críticos, como a integridade do estoque e a atomicidade do fluxo de vendas, garantindo que a aplicação opere de forma consistente sob diversas condições.
- **Aceitação do Utilizador:** A avaliação qualitativa demonstrou que a ferramenta é intuitiva e acessível para empreendedores com diferentes níveis de literacia digital, registrando alta confiança na operação e baixa percepção de complexidade.

### 5.2 Limitações e Análise Crítica

A análise crítica revelou que a principal limitação do projeto reside na ausência de integração com módulos fiscais e tributários. Contudo, essa exclusão foi uma decisão deliberada para preservar a simplicidade e o foco no apoio à decisão financeira interna, diferenciando o sistema de [ERPs](#) genéricos e complexos. Observou-se também que, embora a árvore mercadológica forneça uma organização superior, ela demanda um esforço inicial de configuração que pode ser otimizado em futuras versões.

## 5.3 Trabalhos Futuros

Como perspectivas de evolução para a plataforma, destacam-se:

- **Gestão Automatizada de Suprimentos:** Implementação de mecanismos para o reabastecimento automatizado do catálogo, permitindo o agendamento de reposição de itens em quantidades e intervalos periódicos pré-definidos, visando otimizar o giro de estoque;
- **Módulo de Vendas Recorrentes:** Desenvolvimento de funcionalidades para a automação de processos transacionais voltados a clientes com demanda cíclica, garantindo maior previsibilidade financeira ao empreendedor;
- **Conformidade Fiscal e Tributária:** Estudo para a integração modularizada com serviços de emissão de notas fiscais eletrônicas (NF-e e NFS-e), adequando a plataforma às exigências legais específicas de cada setor de atuação do [MEI \(GS1, 2012\)](#);
- **Expansão para Prestação de Serviços:** Ampliação do escopo funcional do sistema para contemplar a gestão de ordens de serviço e faturamentos baseados em horas ou projetos, adaptando os fluxos de catálogo para além da comercialização de produtos físicos.

Em suma, o sistema proposto demonstrou-se uma solução viável e estratégica, capaz de oferecer clareza financeira e autonomia ao microempreendedor, cumprindo integralmente seu propósito acadêmico e técnico.

# Referências

- ALMEIDA, J. C. de; SILVA, R. d. O. A importância da educação financeira, para a sustentabilidade de pequenos negócios. *REVICOOP*, v. 5, n. 1, 2024. ISSN 2676-0223. Citado 2 vezes nas páginas 16 e 18.
- BECK, K. *Test-driven development: by example*. Boston: Addison-Wesley Professional, 2003. Citado na página 21.
- BRASIL. *Brasil registra abertura de 1,4 milhão de pequenos negócios no primeiro trimestre do ano*. 2025. Secretaria de Comunicação Social. Disponível em: <<https://www.gov.br/secom/pt-br/acompanhe-a-secom/noticias/2025/04/brasil-registra-abertura-de-1-4-milhao-de-pequenos-negocios-no-primeiro-trimestre-do-ano>>. Acesso em: 27 jan. 2026. Citado na página 16.
- CHONG, F.; CARRARO, G. Architecture strategies for catching the long tail. *Microsoft MSDN Library*, 2006. Disponível em: <<https://msdn.microsoft.com/en-us/library/aa479069.aspx>>. Citado na página 21.
- COUTINHO, A. I.; SOUZA, L. C. de. Barreiras que dificultam empresas de pequeno porte de implantarem sistemas erp em comparação as empresas de grande porte. *Interface Tecnológica*, v. 20, n. 1, p. 337–348, 2023. Citado 3 vezes nas páginas 16, 18 e 22.
- GS1. *Global Product Classification (GPC) - Development & Implementation Guide*. 2012. <[https://www.gs1.org/docs/gpc/GPC\\_Development\\_Implementation.pdf](https://www.gs1.org/docs/gpc/GPC_Development_Implementation.pdf)>. Acessado em 15 de janeiro de 2026. Citado 3 vezes nas páginas 20, 38 e 62.
- Inertia.js. *Build single-page apps, without building an API*. 2024. Disponível em: <<https://inertiajs.com>>. Acesso em: 15 jan. 2026. Citado na página 20.
- Meta Platforms Inc. *React: The library for web and native user interfaces*. 2024. Disponível em: <<https://react.dev/>>. Acesso em: 15 jan. 2026. Citado na página 20.
- Microsoft. *TypeScript Documentation*. [S.l.], 2024. Disponível em: <<https://www.typescriptlang.org/docs/>>. Acesso em: 15 jan. 2026. Citado na página 20.
- OTWELL, T. *Laravel: The PHP Framework for Web Artisans*. 2025. Versão 11.x. Disponível em: <<https://laravel.com>>. Acesso em: 15 jan. 2026. Citado na página 19.
- SILVA, F. S. da; BRAGATO, C. G. O processo de informatização do microempreendedor individual: um estudo de uma papelaria no município de colatina - es. *RCMOS - Revista Científica Multidisciplinar O Saber*, v. 1, n. 1, 2025. ISSN 2675-9128. Citado 2 vezes nas páginas 16 e 22.
- TURBAN, E. et al. *Decision Support and Business Intelligence Systems*. 9th. ed. Upper Saddle River: Prentice Hall, 2010. Citado na página 21.

## Apêndices

# APÊNDICE A – Modelagem do banco de dados

Figura 38 – Diagrama de Estrutura do Banco de Dados



Fonte: Elaborado pelo autor utilizando a ferramenta dbdiagram.io.