UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

ALAN PEREIRA DE LIMA

APLICAÇÃO WEB PARA CÁLCULO AUTOMATIZADO DE MARGEM DE LUCRO EM VENDAS ONLINE

GUARAPUAVA

2026

ALAN PEREIRA DE LIMA

APLICAÇÃO WEB PARA CÁLCULO AUTOMATIZADO DE MARGEM DE LUCRO EM VENDAS ONLINE

Web Application for Automated Profit Margin Calculation in Online Sales

Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Tecnólogo em Sistemas para Internet do Curso Superior de Tecnologia em Sistemas para Internet da Universidade Tecnológica Federal do Paraná.

Orientador: Carlos Eduardo Iatskiu

GUARAPUAVA

2026

4.0 Internacional

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

LISTA DE FIGURAS

Figura 1 – Diagrama de casos de uso do sistema 25
Figura 2 – Diagrama de classes do sistema 26
Figura 3 – Diagrama de sequência: sincronização de vendas do Mercado Livre e cálculo de margem 27
Figura 4 – Diagrama de sequência: sincronização de custos via Tiny ERP 28
Figura 5 – Tela de configurações: conexão com o Mercado Livre e gestão de usuários do painel 29
Figura 6 – Tela de dashboard: indicadores consolidados de vendas 30
Figura 7 – Tela de comparativo entre dois períodos 31
Figura 8 – Tela de listagem de vendas do Mercado Livre 32
Figura 9 – Tela de produtos: cadastro e sincronização de custos 33

LISTA DE TABELAS

Tabela 1 – Comparativo entre soluções existentes e a aplicação proposta 14

Tabela 2 – Requisitos funcionais do sistema 21

Tabela 3 – Requisitos não funcionais do sistema 22

LISTA DE QUADROS

Quadro 1 – Ferramentas e tecnologias previstas para o desenvolvimento do sistema 19

LISTA DE ABREVIATURAS E SIGLAS

Siglas

API	*Application Programming Interface*
BI	*Business Intelligence*
CSV	*Comma-Separated Values*
ERP	*Enterprise Resource Planning*
HTTP	*Hypertext Transfer Protocol*
JWT	*JSON Web Token*
MVC	*Model-View-Controller*
MVP	*Minimum Viable Product*
OAuth 2.0	*Open Authorization 2.0*
ORM	*Object-Relational Mapping*
REST	*Representational State Transfer*
RF	Requisito Funcional
RNF	Requisito Não Funcional
SKU	*Stock Keeping Unit*
UML	*Unified Modeling Language*

SUMÁRIO

1 INTRODUÇÃO 7

1.1 Considerações iniciais 7

1.2 Objetivos 8

1.2.1 Objetivo geral 8

1.2.2 Objetivos específicos 8

2 REFERENCIAL TEÓRICO 9

2.1 Gestão de Custos, Precificação e Margem de Lucro em Pequenas Empresas 9

2.2 O Comércio Eletrônico e os Marketplaces no Brasil 11

2.3 Tecnologia Aplicada à Gestão de Vendas: Integração via API e Business Intelligence 12

2.4 Soluções Tecnológicas Relacionadas 13

3 METODOLOGIA 15

3.1 Visão Geral do Trabalho 15

3.2 Técnicas de Levantamento e Priorização de Requisitos 16

3.3 Técnicas de Modelagem do Sistema 17

3.4 Ferramentas e Tecnologias Previstas 17

4 ANÁLISE E PROJETO DO SISTEMA 20

4.1 Escopo do Sistema 20

4.2 Levantamento de Requisitos 20

4.2.1 Requisitos Funcionais 21

4.2.2 Requisitos Não Funcionais 22

4.2.3 Regras de Negócio 22

4.3 Modelagem do Sistema 23

4.3.1 Diagramas de Sequência 26

4.4 Sistema em Produção: Telas de Referência 28

4.4.1 Autenticação e Configurações 28

4.4.2 Dashboard e Indicadores de Vendas 29

4.4.3 Listagem de Vendas do Mercado Livre 31

4.4.4 Produtos e Custos 32

5 CONCLUSÃO 34

REFERÊNCIAS 36

1 INTRODUÇÃO

1.1 Considerações iniciais

A Agrolan Peças Agrícolas é uma microempresa (ME), optante pelo Simples Nacional, fundada em 2017 inicialmente para a venda de peças agrícolas pela internet. Atualmente, a empresa também conta com uma loja física, mas o maior volume de vendas permanece *on-line*: do total de aproximadamente 700 vendas realizadas por mês, cerca de 90% ocorrem no Mercado Livre, 5% no balcão da loja física, 3% no site próprio (Nuvemshop) e 2% na Shopee. O autor deste trabalho atua como gestor da Agrolan, sendo responsável, entre outras atividades, pelo controle financeiro das vendas realizadas em cada canal – contexto que motiva o problema tratado neste trabalho e justifica a delimitação do escopo deste TCC ao Mercado Livre, principal canal de vendas da empresa.

Para lidar com a gestão das vendas realizadas nesses diferentes canais, o autor deste trabalho desenvolveu e mantém, com apoio de ferramentas de inteligência artificial, um sistema de gestão de vendas de uso próprio, hoje em operação real na Agrolan. Esse sistema consolida, entre outras funcionalidades: a emissão de etiquetas de envio compatíveis com a exigência legal de inclusão da chave de acesso da nota fiscal eletrônica (NF-e) de transporte, reformatando o modelo nativo do Mercado Livre – que aproveita mal o espaço da folha – para um layout com três etiquetas por folha A4; o cálculo de frete personalizado, integrado à loja virtual, com prazos e preços configuráveis pelo próprio gestor; e um *dashboard* de vendas, lucro e margem por canal e por dia, que automatiza o cálculo antes feito manualmente em planilha eletrônica – esta última sendo a funcionalidade de interesse direto para este trabalho.

Esse sistema, no entanto, foi construído de forma incremental e informal, conforme as necessidades do dia a dia da empresa foram surgindo, sem levantamento formal de requisitos, documentação das regras de negócio ou modelagem prévia, resultando em uma aplicação monolítica que concentra, em uma única base de código, funcionalidades heterogêneas – como emissão de etiquetas, cálculo de frete regional e integração com múltiplos canais de venda – sem separação clara de responsabilidades nem cobertura de testes automatizados. Este trabalho não se propõe a substituir ou ampliar esse sistema como um todo: seu objetivo é isolar especificamente o problema do cálculo automático da margem de lucro por venda no Mercado Livre e reconstruir essa funcionalidade do zero, desta vez seguindo um processo formal de engenharia de software – com levantamento e priorização de requisitos, definição explícita de regras de negócio, modelagem do sistema antes da implementação e testes automatizados –, de modo a produzir uma solução documentada, testável e avaliável academicamente, e não apenas mais uma funcionalidade incorporada ao sistema existente.

1.2 Objetivos

Esta seção apresenta o objetivo geral e os objetivos específicos deste trabalho.

1.2.1 Objetivo geral

Desenvolver uma aplicação web para o cálculo automático da margem de lucro das vendas realizadas pela Agrolan Peças Agrícolas no Mercado Livre, a partir da integração com a *Application Programming Interface* (API) do marketplace e com a API do Tiny *Enterprise Resource Planning* (ERP), substituindo o controle atualmente realizado de forma manual em planilha eletrônica.

1.2.2 Objetivos específicos

- Desenvolver um mecanismo de sincronização automática dos dados de vendas do Mercado Livre, por meio de integração com a API oficial do marketplace.
- Desenvolver um mecanismo de consulta automática do custo dos produtos vendidos, por meio de integração com a API do Tiny ERP.
- Implementar o cálculo automático do valor líquido, do lucro e da margem de cada venda, considerando as taxas cobradas pelo marketplace e o custo do produto vigente no momento da venda.
- Disponibilizar um *dashboard* web com os indicadores consolidados de vendas, lucro e margem por período selecionado.
- Avaliar o funcionamento do sistema por meio de testes automatizados e de cenários de uso representativos da operação real da empresa.

2 REFERENCIAL TEÓRICO

Este capítulo apresenta a fundamentação teórica que sustenta a proposta de desenvolvimento deste trabalho, reunindo os principais conceitos e estudos relacionados ao problema de precificação e apuração da margem de lucro em pequenas empresas que atuam no comércio eletrônico. A seção 2.1 discute os fundamentos de gestão de custos e formação de preços, evidenciando, por meio de estudos empíricos, os riscos associados ao controle manual desses cálculos em micro e pequenas empresas. Em seguida, a seção 2.2 contextualiza o cenário do comércio eletrônico e dos marketplaces no Brasil, destacando o volume de vendas do setor e o impacto das taxas de comissão cobradas por essas plataformas sobre a margem do vendedor. A seção 2.3 apresenta os conceitos tecnológicos que viabilizam a automação desse processo, com destaque para a integração de sistemas via APIs *Representational State Transfer* (REST) e para o uso de ferramentas de *Business Intelligence* (BI) como apoio à tomada de decisão. Por fim, a seção 2.4 analisa soluções tecnológicas já existentes no mercado que se aproximam, total ou parcialmente, do problema tratado neste trabalho, situando a contribuição da aplicação proposta em relação a elas.

2.1 Gestão de Custos, Precificação e Margem de Lucro em Pequenas Empresas

O avanço do comércio eletrônico no Brasil tem exigido das micro e pequenas empresas uma gestão financeira cada vez mais criteriosa, especialmente no que se refere à formação de preços e ao acompanhamento da margem de lucro obtida em cada venda. Segundo Dubois, Kulpa e Souza (2019), os preços de venda praticados por uma empresa quase sempre devem refletir a totalidade dos custos envolvidos em sua elaboração, de modo que a contabilização correta de todos os insumos utilizados é o que permite à empresa medir o seu grau de eficiência e evitar distorções significativas no preço final dos bens e serviços comercializados. Os autores destacam ainda que falhas na gestão e na formação dos preços de venda estão entre as causas mais apontadas por economistas, administradores e contadores para o encerramento precoce das atividades de pequenas empresas, o que reforça a importância de um controle preciso de custos e margens desde os primeiros anos de operação do negócio.

Para fins deste trabalho, adotam-se as definições de Dubois, Kulpa e Souza (2019): o custo corresponde aos gastos diretamente relacionados à aquisição do bem vendido (por exemplo, o valor pago ao fornecedor por uma unidade de produto), enquanto a despesa corresponde aos gastos necessários à operação do negócio, mas não vinculados diretamente ao produto (como taxas administrativas e tributos sobre a venda). A partir desses conceitos, o *markup* é o índice aplicado sobre o custo do produto para a formação do preço de venda; a margem de contribuição corresponde à diferença entre o preço de venda e os custos e despesas variáveis associados a essa venda; e a margem de lucro é o percentual que o lucro líquido representa

sobre o preço de venda. São esses os conceitos que o sistema desenvolvido neste trabalho automatiza para cada venda realizada em marketplaces.

Essas relações podem ser expressas pelas seguintes fórmulas:

$$\text{Lucro Líquido} = \text{Preço de Venda} - \text{Custo} - \text{Despesa}$$

$$\text{Markup} = \frac{\text{Preço de Venda}}{\text{Custo}}$$

$$\text{Margem de Contribuição} = \text{Preço de Venda} - (\text{Custos Variáveis} + \text{Despesas Variáveis})$$

$$\text{Margem de Lucro} = \frac{\text{Lucro Líquido}}{\text{Preço de Venda}} \times 100\%$$

Estudos empíricos evidenciam que essa dificuldade não é apenas teórica. Purificação *et al.* (2020) identificaram, junto a gestores de micro e pequenas empresas do interior paulista, que 66% deles relatam dificuldade em mensurar corretamente seus custos, e mais da metade não utiliza qualquer método formal de precificação, mesmo revisando os preços periodicamente. Resultado semelhante foi encontrado por Junior *et al.* (2005) em um estudo de caso em um supermercado de médio porte: antes da implantação de um controle de custos estruturado, o estabelecimento operava com prejuízo líquido de 1% sobre as vendas e ponto de equilíbrio operacional (sem considerar as despesas financeiras) em 83% do faturamento; após passar a monitorar diariamente seus custos e despesas financeiras, antes “invisíveis” na gestão do negócio, como os juros do cheque especial e o desconto de cheques, a empresa reduziu o ponto de equilíbrio para 61% e passou a operar com lucro líquido de 7% sobre as vendas em apenas dois meses.

Esse cenário se repete mesmo em empresas com bom desempenho histórico. Carvalho *et al.* (2008), ao estudarem uma pequena empresa metalúrgica de produção sob encomenda no Rio de Janeiro, constataram que, apesar dos resultados financeiros satisfatórios, a formação de preços da empresa dependia fortemente da intuição do gestor, e não de um sistema de custeio formalizado, recomendando a adoção do custeio variável e do rateio de custos por atividade como forma de qualificar essa decisão. Como alternativa ao markup, Pissinati e Dias (2022) defendem o uso da margem de contribuição como métrica de precificação, por permitir a formação de preços competitivos sem desconsiderar os custos fixos e variáveis do negócio, favorecendo tanto a sustentabilidade financeira quanto a tomada de decisão gerencial.

O controle manual desses cálculos, frequentemente realizado em planilhas eletrônicas, também é apontado como fator de risco. Maia, Melo e Dias (2023), em estudo de caso em uma pequena empresa do ramo alimentício, verificaram que a inexistência de um controle de custos

automatizado levava a erros na apuração do preço praticado, sendo necessária a criação de uma planilha de custos estruturada para corrigir o problema. Situação análoga motiva o presente trabalho, que propõe automatizar esse cálculo para cada venda realizada em marketplaces. Zahaikevitch *et al.* (2018) reforçam que o uso da contabilidade gerencial como ferramenta de apoio à decisão é reconhecido pelos próprios gestores de micro e pequenas empresas como fundamental para agilizar decisões com base em informações financeiras precisas, ainda que sua adoção efetiva permaneça limitada nesse porte de empresa.

Embora os estudos citados tratem de segmentos distintos, como supermercado, indústria metalúrgica e comércio alimentício, o desafio de precificar corretamente sem incorrer em prejuízo é igualmente relevante, ou ainda mais crítico, no comércio eletrônico, em que cada venda está sujeita a taxas de intermediação específicas do canal, como discutido na seção a seguir.

2.2 O Comércio Eletrônico e os Marketplaces no Brasil

O comércio eletrônico brasileiro consolidou-se, na última década, como um dos principais canais de venda para pequenos e médios negócios. Segundo levantamento da Mercado&Consumo (2026), o setor movimentou R$ 125 bilhões no primeiro semestre de 2026, crescimento de 24,37% em relação ao mesmo período de 2025 (R$ 100,5 bilhões), com projeção de faturamento anual superior a R$ 260 bilhões e 460,87 milhões de pedidos até o fim do ano. Parte relevante desse volume é intermediada por marketplaces, plataformas que reúnem múltiplos vendedores em um único ambiente de compra, que se tornaram, para muitas micro e pequenas empresas, o principal ou até único canal de vendas *on-line* (Sebrae, 2026).

Essa intermediação, no entanto, tem um custo direto sobre a margem do vendedor. De acordo com Bling (2026b), Mercado Livre (2026), o Mercado Livre, maior marketplace da América Latina, cobra comissões que variam entre 10% e 19% do valor do produto, a depender da categoria e do tipo de anúncio (clássico ou premium), acrescidas de uma tarifa fixa de até R$ 6,75 por unidade vendida em produtos de menor valor. Em um exemplo apresentado pela fonte, um produto vendido por R$ 75,00 no plano clássico, com comissão de 12%, resulta em um repasse de apenas R$ 59,25 ao vendedor. Essa é uma redução de 21% sobre o valor bruto da venda, antes mesmo de descontados os custos de aquisição do produto e demais despesas operacionais. Esse tipo de taxa variável, somado aos custos de frete, impostos e ao próprio custo do produto, torna o cálculo manual da margem de lucro suscetível a erros e cada vez mais inviável à medida que o volume de vendas cresce, o que justifica a busca por soluções que automatizem esse processo.

2.3 Tecnologia Aplicada à Gestão de Vendas: Integração via API e Business Intelligence

A automatização do cálculo de custos e margens em um cenário de múltiplos canais de venda depende da capacidade de integrar sistemas distintos, como a plataforma de vendas, o sistema de gestão de estoque e custos (ERP) e a própria aplicação de análise. Essa integração é viabilizada, do ponto de vista técnico, pelo uso de APIs do tipo REST. Segundo Junior e Santos (2019) e Fielding (2000), a arquitetura REST permite a comunicação entre subsistemas e sistemas externos por meio de requisições *Hypertext Transfer Protocol* (HTTP), possibilitando que aplicações distintas troquem dados de forma padronizada e independente, sem que a alteração de um módulo comprometa o funcionamento dos demais. No caso estudado pelos autores, por exemplo, esse princípio foi observado em um sistema com *back-end* em Node.js e *front-end* separado em Angular. Esse mesmo princípio de separação entre *back-end* e *front-end*, consumindo APIs externas, é o que possibilita que uma aplicação web obtenha, por exemplo, os dados de vendas diretamente da API pública do Mercado Livre (Mercado Livre Developers, 2026) e os dados de custo de produtos diretamente da API do sistema de gestão Tiny (Tiny ERP, 2026), sem a necessidade de inserção manual dessas informações.

O acesso a essas APIs externas depende, ainda, de mecanismos de autorização e autenticação adequados. O *Open Authorization 2.0* (OAuth 2.0) é um protocolo de autorização que permite que uma aplicação obtenha acesso limitado aos dados de um usuário em outro serviço, sem a necessidade de compartilhar suas credenciais, por meio de tokens de acesso emitidos por um servidor de autorização (HARDT, 2012); é esse o mecanismo utilizado pelo Mercado Livre para autorizar o acesso da aplicação aos dados de vendas do vendedor. Já o *JSON Web Token* (JWT) é um formato compacto e autocontido para a representação de informações entre duas partes, assinado digitalmente para garantir sua integridade, amplamente utilizado para a autenticação de usuários e a proteção de rotas em aplicações web (JONES; BRADLEY; SAKIMURA, 2015), sendo o formato adotado pela própria aplicação proposta para autenticar seus usuários.

Uma vez integrados, esses dados precisam ser apresentados de forma que apoiem a tomada de decisão do gestor, papel desempenhado pelas ferramentas de BI, conjunto de técnicas, tecnologias e processos utilizados para coletar, integrar, analisar e apresentar informações que apoiem a tomada de decisão organizacional (TURBAN; SHARDA; DELEN, 2009). Silva e Mesquita (2024), em trabalho de conclusão de curso que avaliou a adoção de dashboards de BI em micro e pequenas empresas, constatou que a substituição de planilhas eletrônicas por painéis interativos reduziu o tempo de análise dos dados, aumentou a velocidade e a precisão das decisões tomadas pelos gestores e eliminou a dependência de registros manuais. Esse achado corrobora a proposta do presente trabalho, na qual um dashboard alimentado automaticamente pelas APIs de vendas e de custos substitui o controle manual anteriormente realizado em planilha eletrônica, permitindo a visualização consolidada de vendas, custos, taxas e margem de lucro por pedido.

2.4 Soluções Tecnológicas Relacionadas

Não se trata de um problema inédito: diferentes soluções, com escopos e públicos distintos, já buscam auxiliar o pequeno empreendedor a controlar custos e vender em múltiplos canais. Sistemas de gestão empresarial (ERP) como o Tiny e o Bling concentram funções de emissão de nota fiscal, controle de estoque e cadastro de custo dos produtos, e alguns deles oferecem sincronização com marketplaces (Turbo Partners, 2026). O foco desses sistemas está historicamente na gestão fiscal e logística do negócio: o Tiny ERP não oferece um relatório nativo de margem por venda, dependendo de ferramentas externas de BI para esse tipo de análise, enquanto o Bling passou recentemente a oferecer, nos planos mais altos, um painel de margem por pedido restrito a vendas do Mercado Livre, que importa automaticamente as taxas da plataforma (Bling, 2026a); o recurso, porém, exige configuração manual de custo e alíquota de imposto por loja, não reprocessa pedidos antigos e não integra dados de outros canais ou ERPs. Ferramentas como a planilha de precificação para marketplace disponibilizada gratuitamente pelo Sebrae (Sebrae, 2026) aproximam-se mais do problema tratado neste trabalho, ao oferecerem um modelo de cálculo de preço de venda que já considera comissões de marketplace, mas dependem do preenchimento manual dos dados de cada produto, o que limita sua aplicação a um número reduzido de itens e pedidos. Já ferramentas voltadas à análise de concorrência em marketplaces, como a Nubimetrics (Nubimetrics, 2026), concentram-se no comportamento de mercado e dos concorrentes, sem tratar do custo interno do vendedor nem calcular sua margem de lucro efetiva.

A Tabela 1 sintetiza essa comparação.

Tabela 1 – Comparativo entre soluções existentes e a aplicação proposta

Sistema	Foco principal	Integração via API com marketplace	Cálculo automático de margem por venda	Limitação principal
Tiny ERP	Gestão fiscal, estoque e custo dos produtos	Sim (parcial)	Não	Não calcula margem por venda; requer BI externo
Bling	Gestão fiscal, estoque e custo dos produtos	Sim (parcial)	Parcial (só Mercado Livre)	Restrito a planos superiores; não retroativo; não integra outros ERPs
Planilha Sebrae	Cálculo de preço de venda para marketplace	Não	Sim (manual)	Depende do preenchimento manual por produto
Nubimetrics	Análise de concorrência e mercado	Sim (leitura de anúncios)	Não	Não considera o custo interno do vendedor
Planilha eletrônica (uso manual)	Controle financeiro genérico	Não	Sim (manual)	Sujeita a erro humano; não escalável com o volume de pedidos
Aplicação proposta neste trabalho	Dashboard de vendas com cálculo automático de margem	Sim (via API)	Sim (automático)	Restrita ao Mercado Livre e ao Tiny ERP

Fonte: Elaborado pelo autor com base em Turbo Partners (2026), Sebrae (2026), Nubimetrics (2026) e Bling (2026a).

A análise comparativa evidencia uma lacuna entre as soluções existentes: o Tiny ERP cobre bem a gestão fiscal e de estoque, mas não calcula a margem de lucro por venda; o Bling já oferece um cálculo parcial de margem, mas restrito a vendas do Mercado Livre, aos planos superiores da plataforma e sem integração com dados de outros ERPs; ferramentas de precificação como a planilha do Sebrae calculam a margem, mas exigem preenchimento manual, o que inviabiliza sua aplicação em um volume elevado de pedidos; e ferramentas de análise de mercado como a Nubimetrics não tratam do custo interno do vendedor. Essa lacuna existe entre a integração automatizada de dados, presente nos ERPs, e o cálculo automático da margem de lucro que combine, de forma aberta a múltiplos canais e sistemas, custo do produto e taxas do canal de venda. É exatamente nessa lacuna que se posiciona a aplicação web proposta neste trabalho, que integra, via API, os dados de vendas do marketplace e os dados de custo do ERP para calcular automaticamente a margem de lucro de cada venda realizada.

3 METODOLOGIA

Este capítulo apresenta a metodologia adotada para o desenvolvimento deste trabalho, descrevendo a natureza da pesquisa, o sistema proposto e a forma como as etapas de análise, modelagem e implementação serão conduzidas. A seção 3.1 apresenta uma visão geral do trabalho, caracterizando o tipo de pesquisa realizada e descrevendo, em linhas gerais, o sistema a ser desenvolvido. Em seguida, a seção 3.2 descreve as técnicas que serão utilizadas para o levantamento e a priorização dos requisitos do sistema. A seção 3.3 apresenta as técnicas de modelagem previstas para representar a estrutura e o comportamento do sistema. Por fim, a seção 3.4 relaciona as ferramentas e tecnologias previstas para o desenvolvimento da aplicação proposta.

3.1 Visão Geral do Trabalho

Este trabalho caracteriza-se como uma pesquisa aplicada, de natureza tecnológica (GIL, 2002), voltada ao desenvolvimento de um sistema web que automatize o cálculo da margem de lucro sobre as vendas realizadas por uma pequena empresa em um *marketplace*. Diferentemente de uma pesquisa puramente teórica, a proposta centra-se na concepção, no planejamento e na implementação de uma solução computacional capaz de substituir um processo até então realizado de forma manual pelo próprio autor deste trabalho, na condição de gestor da empresa envolvida.

A definição de requisitos, regras de negócio e critérios de priorização parte da vivência prática do problema pelo próprio autor deste trabalho, e não de instrumentos estatísticos ou experimentais formais. Quanto aos procedimentos técnicos, o trabalho classifica-se como um projeto de desenvolvimento de software, estruturado nas etapas de levantamento de requisitos, modelagem do sistema, implementação e avaliação dos resultados obtidos, conduzidas de forma iterativa: a cada funcionalidade implementada, o sistema passa a ser utilizado na operação real da Agrolan e é ajustado conforme necessário antes de se avançar para a próxima. A etapa de avaliação consiste na aplicação do sistema a uma amostra de pedidos de vendas reais, de aproximadamente 50 vendas referentes a um mês de operação, comparando a margem de lucro calculada automaticamente pelo sistema com a obtida pelo processo manual anteriormente realizado em planilha eletrônica, bem como o tempo despendido em cada um dos dois processos.

Conforme discutido no Capítulo 2, o controle manual do cálculo de custos, taxas e margem de lucro em planilhas eletrônicas é uma prática comum entre pequenas empresas que atuam em *marketplaces*, estando sujeito a erros e à dificuldade de escalar com o aumento do volume de pedidos. A Tabela 1 evidencia, ainda, que as soluções tecnológicas já existentes no mercado não cobrem de forma integrada a automação desse cálculo. É nesse contexto que se insere o sistema proposto neste trabalho: uma aplicação web que consulta automaticamente,

via API, os dados de vendas do Mercado Livre e os dados de custo dos produtos cadastrados no ERP Tiny, calculando de forma automática a margem de lucro de cada venda realizada e apresentando esses dados de forma consolidada em um *dashboard*.

3.2 Técnicas de Levantamento e Priorização de Requisitos

O levantamento de requisitos deste trabalho parte de uma característica particular do problema tratado: o autor deste trabalho é, simultaneamente, o desenvolvedor do sistema e o gestor da empresa para a qual ele será construído, tendo vivenciado diretamente, ao longo da operação da Agrolan, as dificuldades relacionadas ao cálculo manual da margem de lucro por venda. Essa vivência prática constitui a principal fonte de requisitos do sistema, complementada pela análise de outras fontes de informação.

Serão utilizadas três técnicas principais de levantamento de requisitos. A primeira consiste na observação e experiência direta do autor sobre o processo manual anteriormente realizado em planilha eletrônica, identificando quais dados eram consultados, em quais fontes e quais cálculos eram efetuados a cada venda. A segunda consiste na análise do sistema de gestão de vendas já em produção na empresa (apresentado no Capítulo 1), utilizado como base informal para identificar quais funcionalidades relacionadas ao cálculo de margem já se mostraram necessárias na prática, sem que isso implique a replicação integral desse sistema no escopo deste trabalho. A terceira consiste na análise da documentação oficial das APIs externas que serão integradas – a API do Mercado Livre e a API do Tiny ERP –, de modo a compreender quais dados estão disponíveis, em qual formato e sob quais limitações técnicas, o que impacta diretamente na definição do que é tecnicamente viável implementar.

A partir dessas fontes, os requisitos levantados serão classificados em requisitos funcionais, que descrevem as ações que o sistema deve executar (como consultar vendas, calcular margem e exibir o *dashboard*), e requisitos não funcionais, que estabelecem critérios de qualidade (como tempo de resposta na consulta às APIs externas e segurança no acesso aos dados, viabilizada por meio de autenticação via JWT).

A priorização dos requisitos levantados será conduzida por meio do método MoSCoW, amplamente utilizado em projetos de desenvolvimento de software (CLEGG; BARKER, 1994) para organizar o escopo de acordo com o grau de importância de cada requisito para a versão inicial do sistema. O método classifica os requisitos em quatro categorias: *Must Have*, correspondente às funcionalidades essenciais, sem as quais o sistema não cumpre seu propósito – como o cálculo automático de margem por venda a partir dos dados do Mercado Livre e do Tiny ERP; *Should Have*, relativo a funcionalidades importantes, mas que não inviabilizam uma primeira versão funcional; *Could Have*, relacionado a funcionalidades desejáveis que poderão ser incorporadas conforme a disponibilidade de tempo; e *Won't Have*, que reúne funcionalidades explicitamente fora do escopo deste trabalho, como a integração com outros canais de venda. Por fim, os requisitos priorizados serão validados junto aos demais usuários do sistema na Agro-

lan, que utilizam o *dashboard* no dia a dia da operação e podem apontar ajustes de prioridade a partir do uso real da ferramenta.

3.3 Técnicas de Modelagem do Sistema

A modelagem do sistema tem como propósito representar, antes da etapa de implementação, a estrutura de dados, o comportamento e os principais fluxos de interação da aplicação proposta, servindo de base para o desenvolvimento e de registro formal das decisões de projeto tomadas.

Para representar a interação entre o usuário e o sistema, será utilizado o Diagrama de Casos de Uso, em notação *Unified Modeling Language* (UML) (GUEDES, 2018), contemplando as principais funcionalidades previstas: autenticação do usuário, consulta ao *dashboard* de vendas, sincronização dos dados de vendas via API do Mercado Livre e sincronização dos dados de custo via API do Tiny ERP.

A estrutura de dados do sistema será representada por meio de um Diagrama de Classes, correspondente ao modelo de dados definido no esquema do *Object-Relational Mapping* (ORM) Prisma e persistido no banco de dados SQLite. Esse diagrama deverá representar as principais entidades do sistema – como vendas, produtos, custos e usuários – e os relacionamentos entre elas, servindo como referência tanto para a implementação do banco de dados quanto para a compreensão da estrutura de informações manipuladas pela aplicação.

O comportamento dinâmico do sistema, em especial o fluxo de integração entre a interface do usuário, o servidor de aplicação e as APIs externas, será representado por meio de Diagramas de Sequência. Esses diagramas deverão detalhar, por exemplo, o fluxo de requisição dos dados de venda ao Mercado Livre, a requisição do custo do produto correspondente ao Tiny ERP e o cálculo da margem de lucro a partir desses dados, evidenciando a ordem das trocas de mensagens entre os componentes do sistema e eventuais pontos de tratamento de erro, como falhas de comunicação com as APIs externas.

Complementarmente, a interface do sistema será representada por meio de protótipos de tela de alta fidelidade, elaborados com apoio de uma ferramenta de prototipação baseada em inteligência artificial (detalhada na seção 3.4), permitindo validar a disposição dos elementos visuais e o fluxo de navegação do *dashboard* antes do início da implementação.

3.4 Ferramentas e Tecnologias Previstas

A definição das ferramentas e tecnologias previstas para o desenvolvimento do sistema busca atender aos requisitos levantados, priorizando tecnologias amplamente adotadas pelo mercado, com boa documentação e compatibilidade de integração entre si.

Para a camada de servidor (*back-end*), será utilizado o NestJS, framework para Node.js baseado em TypeScript, que organiza a aplicação em módulos (*modules*), compostos por controladores (*controllers*) e provedores de serviço (*providers*/*services*), seguindo uma estrutura inspirada no padrão *Model-View-Controller* (MVC); como a camada de apresentação é inteiramente separada no *front-end*, não há camada de *View* no *back-end*, e o NestJS fornece uma estrutura padronizada para a criação de APIs REST. A persistência dos dados será gerenciada pelo Prisma, um ORM para TypeScript que abstrai o acesso ao banco de dados por meio de um esquema declarativo de modelos, utilizado em conjunto com o SQLite como sistema de gerenciamento de banco de dados – escolha que privilegia a simplicidade de configuração e é suficiente para o volume de dados e o número de usuários previstos para o sistema.

Na camada de apresentação (*front-end*), será utilizado o React em conjunto com o Vite como ferramenta de *build*, compondo uma aplicação inteiramente executada no navegador (*client-side*), separada do *back-end* e integrada a ele exclusivamente por meio de chamadas à API. Para a construção da interface, será utilizada a biblioteca de componentes Mantine UI, o que reduz a necessidade de estilização manual dos elementos visuais e contribui para a padronização do *dashboard*.

A autenticação dos usuários do sistema será implementada por meio de JWT, seguindo os padrões e recomendações oficiais do próprio NestJS para esse tipo de mecanismo. A documentação da API desenvolvida será gerada com o Swagger, permitindo consultar e testar os *endpoints* disponibilizados pelo *back-end* durante o desenvolvimento.

Para a elaboração dos diagramas de modelagem apresentados na seção 3.3, será utilizada a ferramenta Mermaid, que permite a criação dos diagramas por meio de notação textual, favorecendo o versionamento junto ao código-fonte do projeto. Já os protótipos de interface do sistema serão elaborados com o apoio do Google Stitch, ferramenta do Google que utiliza inteligência artificial para gerar telas e componentes de interface a partir de descrições em linguagem natural, o que agiliza a criação de protótipos de alta fidelidade sem a necessidade de desenho manual de cada elemento visual. Nesta etapa do trabalho, a seção 4.4 apresenta as telas do sistema atualmente em produção na empresa, utilizadas como referência para o levantamento dos requisitos; os protótipos de interface do sistema proposto, elaborados no Google Stitch, serão desenvolvidos em etapa posterior do projeto.

O controle de versão do código-fonte será realizado com Git, com repositório remoto hospedado no GitHub, organizado como um *monorepo* com as pastas `backend/` e `frontend/` separadas. Para a validação do sistema, serão utilizados testes automatizados: no *back-end*, o Jest, framework de testes já integrado à estrutura padrão de projetos NestJS; no *front-end*, o Vitest em conjunto com a React Testing Library, para a validação de componentes e fluxos de interface.

Com o objetivo de sintetizar as ferramentas e tecnologias apresentadas nesta seção, o Quadro 1 relaciona cada uma delas, a versão prevista, o endereço da documentação oficial e a finalidade no sistema.

Quadro 1 – Ferramentas e tecnologias previstas para o desenvolvimento do sistema

Ferramenta	**Versão**	**Disponível em**	**Finalidade**
NestJS	12.3.2	https://docs.nestjs.com	Framework do *back-end*
Prisma	7.10.1	https://www.prisma.io/docs	ORM de acesso ao banco de dados
React	19.3.1	https://react.dev	Biblioteca do *front-end*
Vite	8.3.5	https://vite.dev	Ferramenta de *build* do *front-end*
Mantine UI	9.7.4	https://mantine.dev	Biblioteca de componentes de interface
Swagger	12.0.5	https://swagger.io	Documentação da API
Jest	30.5.2	https://jestjs.io	Testes automatizados do *back-end*
Vitest	5.0.5	https://vitest.dev	Testes automatizados do *front-end*
Google Stitch	–	https://stitch.withgoogle.com	Prototipação de interface com apoio de inteligência artificial
Mermaid	12.2.0	https://mermaid.js.org	Diagramas de modelagem do sistema
Git / GitHub	–	https://git-scm.com	Controle de versão do código-fonte

Fonte: O autor (2026).

4 ANÁLISE E PROJETO DO SISTEMA

Este capítulo apresenta a análise e o projeto do sistema a ser desenvolvido, detalhando o escopo, os requisitos, a modelagem e os protótipos de interface previstos para a aplicação. A seção 4.1 delimita o escopo do sistema, caracterizando-o como um produto mínimo viável. Em seguida, a seção 4.2 apresenta os requisitos funcionais e não funcionais levantados a partir das técnicas descritas no Capítulo 3. A seção 4.3 apresenta a modelagem do sistema, por meio dos diagramas de casos de uso, de classes e de sequência. Por fim, a seção 4.4 apresenta as telas do sistema atualmente em produção na empresa, que serviram de referência para o levantamento dos requisitos e que serão evoluídas no sistema proposto.

4.1 Escopo do Sistema

O sistema proposto corresponde a um produto mínimo viável (*Minimum Viable Product* (MVP)), contemplando as funcionalidades estritamente necessárias para automatizar o cálculo da margem de lucro das vendas realizadas no Mercado Livre.

O MVP contempla três grupos de funcionalidades: um dashboard com os principais indicadores de vendas (quantidade de vendas, valor bruto, valor líquido, lucro e margem), calculados a partir da integração automática com a API do Mercado Livre; uma listagem detalhada das vendas realizadas, produto a produto, com os respectivos valores de bruto, taxas, líquido, custo, lucro e margem; e o cadastro e a sincronização do custo dos produtos, obtidos manualmente ou por meio da API do Tiny ERP, utilizados como base para o cálculo automático da margem.

Ficam explicitamente fora do escopo deste trabalho os itens a seguir, classificados com prioridade *Won't Have* segundo o método MoSCoW:

- Integração com outros marketplaces além do Mercado Livre.
- Emissão de documentos fiscais.
- Controle de estoque.
- Cálculo e detalhamento de fretes e impostos.
- Sugestão automática de preço de venda, cujo cálculo correto exigiria considerar as taxas variáveis de cada categoria do marketplace.

4.2 Levantamento de Requisitos

A partir das técnicas de levantamento descritas no Capítulo 3 – em especial da vivência prática do autor como gestor da empresa e da análise do sistema já em produção –, foram levantados os requisitos funcionais (Requisito Funcional (RF)) e não funcionais (Requisito

Não Funcional (RNF)) do sistema, apresentados nas subseções 4.2.1 e 4.2.2, respectivamente. Cada requisito foi priorizado por meio do método MoSCoW, considerando o escopo de MVP definido na seção 4.1.

4.2.1 Requisitos Funcionais

A Tabela 2 apresenta os requisitos funcionais levantados para o sistema, numerados de RF01 a RF16.

Tabela 2 – Requisitos funcionais do sistema

ID	Descrição	Prioridade
RF01	O sistema deve permitir que o usuário se autentique por meio de e-mail e senha.	Must Have
RF02	O sistema deve permitir o cadastro e a remoção de usuários do painel, cada um com e-mail e senha próprios.	Should Have
RF03	O sistema deve permitir conectar uma conta do Mercado Livre por meio de autenticação OAuth 2.0, armazenando as credenciais de acesso (token de acesso e refresh token) e renovando-as automaticamente quando expiradas.	Must Have
RF04	O sistema deve sincronizar, sob solicitação do usuário, os pedidos e itens vendidos no Mercado Livre em um período informado.	Must Have
RF05	O sistema deve calcular, para cada item vendido, o valor líquido da venda a partir do valor bruto e das taxas informadas pelo Mercado Livre.	Must Have
RF06	O sistema deve consultar o custo do produto correspondente, pelo *Stock Keeping Unit* (SKU), para calcular o lucro e a margem de cada venda.	Must Have
RF07	O sistema deve exibir, em um dashboard, os indicadores consolidados de quantidade de vendas, valor bruto, valor líquido, lucro e margem de um período selecionado.	Must Have
RF08	O sistema deve permitir filtrar o período exibido no dashboard por meio de atalhos predefinidos (hoje, ontem, 7 dias, mês, mês anterior) ou de um intervalo de datas customizado.	Must Have
RF09	O sistema deve exibir um gráfico do faturamento líquido por dia e um gráfico da distribuição entre lucro, custo e taxas no período selecionado.	Should Have
RF10	O sistema deve permitir comparar os indicadores de vendas de dois períodos distintos, lado a lado.	Should Have
RF11	O sistema deve exibir uma listagem das vendas realizadas, produto a produto, com os respectivos valores de bruto, taxas, líquido, custo, lucro e margem.	Must Have
RF12	O sistema deve permitir buscar e filtrar a listagem de vendas por produto, SKU ou identificador do pedido.	Could Have
RF13	O sistema deve permitir exportar a listagem de vendas em formato *Comma-Separated Values* (CSV).	Could Have
RF14	O sistema deve permitir o cadastro manual do custo de um produto, informando SKU, custo e título do anúncio.	Must Have

(continua)

Tabela 2 – Requisitos funcionais do sistema

(continuação)

ID	Descrição	Prioridade
RF15	O sistema deve permitir a sincronização dos custos dos produtos cadastrados a partir da API do Tiny ERP.	Must Have
RF16	O sistema deve permitir buscar, editar e remover produtos cadastrados.	Should Have

Fonte: Elaborado pelo autor

4.2.2 Requisitos Não Funcionais

A Tabela 3 apresenta os requisitos não funcionais levantados para o sistema, numerados de RNF01 a RNF08.

Tabela 3 – Requisitos não funcionais do sistema

ID	Descrição	Prioridade
RNF01	O sistema deve responder às consultas do dashboard e da listagem de vendas em até 3 segundos, mesmo com um volume de até 2.000 produtos cadastrados e 5.000 itens de venda.	Must Have
RNF02	O acesso ao sistema deve ser restrito a usuários autenticados, por meio de token JWT.	Must Have
RNF03	O sistema deve ser acessível por meio de navegador web, sem necessidade de instalação local.	Must Have
RNF04	A interface do sistema deve seguir um padrão visual consistente entre as telas, por meio do uso de uma biblioteca de componentes.	Should Have
RNF05	O sistema deve manter cobertura de testes automatizados para as regras de negócio críticas (cálculo de valor líquido, lucro e margem), de modo a facilitar a manutenção e evitar regressões.	Should Have
RNF06	O sistema deve armazenar as senhas dos usuários utilizando um algoritmo de hash seguro (por exemplo, bcrypt ou argon2), nunca em texto plano.	Must Have
RNF07	O sistema deve armazenar de forma segura os tokens de acesso e de renovação (*refresh token*) da API do Mercado Livre.	Must Have
RNF08	O sistema deve respeitar os limites de requisição (*rate limit*) das APIs do Mercado Livre e do Tiny ERP, evitando o bloqueio da integração.	Must Have

Fonte: Elaborado pelo autor

4.2.3 Regras de Negócio

As regras a seguir complementam os requisitos funcionais e não funcionais, definindo o comportamento esperado do sistema diante de situações que impactam o cálculo dos indicadores financeiros e a interpretação dos dados apresentados ao usuário.

RN01 O valor líquido de uma venda é calculado subtraindo-se do valor bruto as taxas e comissões cobradas pelo marketplace. O lucro corresponde ao valor líquido subtraído do custo do produto vendido. A margem é calculada como a razão entre o lucro e o valor líquido, expressa em percentual.

RN02 Quando o produto vendido não possui custo cadastrado no momento da venda, o sistema considera o custo como zero para fins de cálculo. Nesse caso, o lucro exibido iguala-se ao valor líquido e a margem é apresentada como 100%, o que constitui uma limitação conhecida do cálculo automático, cabendo ao usuário cadastrar o custo para obter indicadores precisos.

RN03 A margem de uma venda é calculada e fixada no momento da sincronização do pedido, com base no custo do produto vigente naquele instante. Alterações posteriores no custo cadastrado de um produto, decorrentes por exemplo de uma nova sincronização com o Tiny ERP, não alteram retroativamente a margem de vendas já sincronizadas.

RN04 Pedidos cancelados ou devolvidos no Mercado Livre são excluídos do cálculo dos indicadores de valor líquido, lucro e margem, bem como da listagem de vendas, sendo contabilizados apenas os pedidos aprovados. A quantidade e o valor bruto dos pedidos cancelados ou devolvidos no período são exibidos separadamente no dashboard, como indicador informativo, sem impactar os demais indicadores.

RN05 Para fins dos indicadores do dashboard, a “quantidade de vendas” corresponde ao número de pedidos aprovados. A listagem de vendas (RF11), por sua vez, é apresentada por item, podendo um mesmo pedido conter mais de um item.

RN06 O sistema não realiza sugestão automática de preço de venda. Esse recurso é classificado como *Won’t Have* no escopo deste projeto (seção 4.1), uma vez que seu cálculo correto exigiria considerar as taxas variáveis de cada categoria do marketplace.

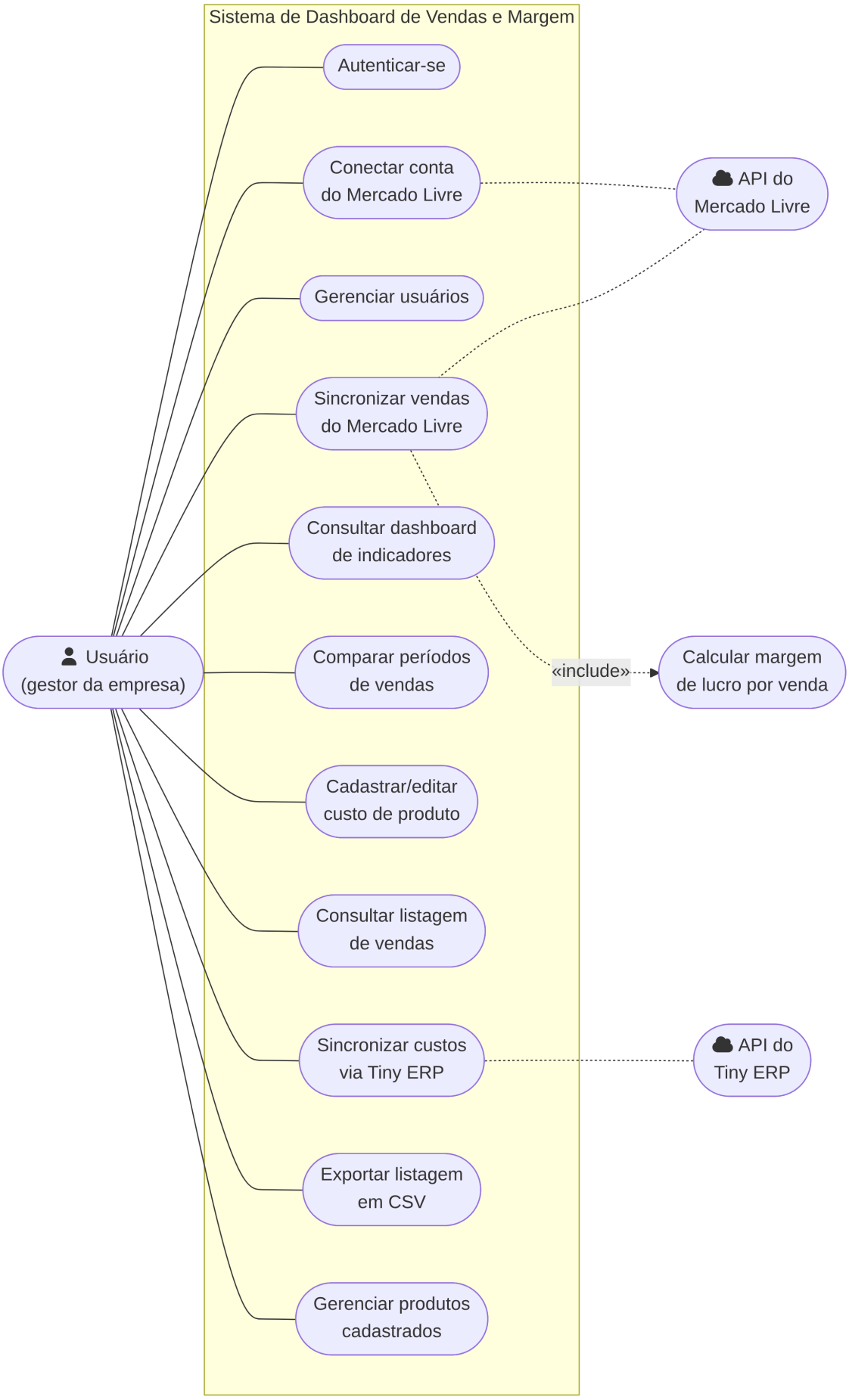
4.3 Modelagem do Sistema

A modelagem do sistema foi elaborada a partir dos requisitos levantados na seção 4.2, seguindo as técnicas descritas no Capítulo 3. Os diagramas apresentados a seguir foram elaborados com a ferramenta Mermaid, utilizada para a modelagem do sistema.

A Figura 1 apresenta o diagrama de casos de uso do sistema, em notação UML, contemplando as principais funcionalidades previstas para o ator primário *Usuário* (gestor da empresa), além dos atores secundários *API do Mercado Livre* e *API do Tiny ERP*, representando os sistemas externos com os quais a aplicação se integra. Destaca-se que o caso de uso “Sincronizar vendas do Mercado Livre” inclui o caso de uso “Calcular margem de lucro por venda”, uma vez que, conforme detalhado na subseção 4.3.1, o cálculo da margem é realizado automaticamente

no momento da sincronização dos dados de venda, com base no custo do produto vigente naquele instante.

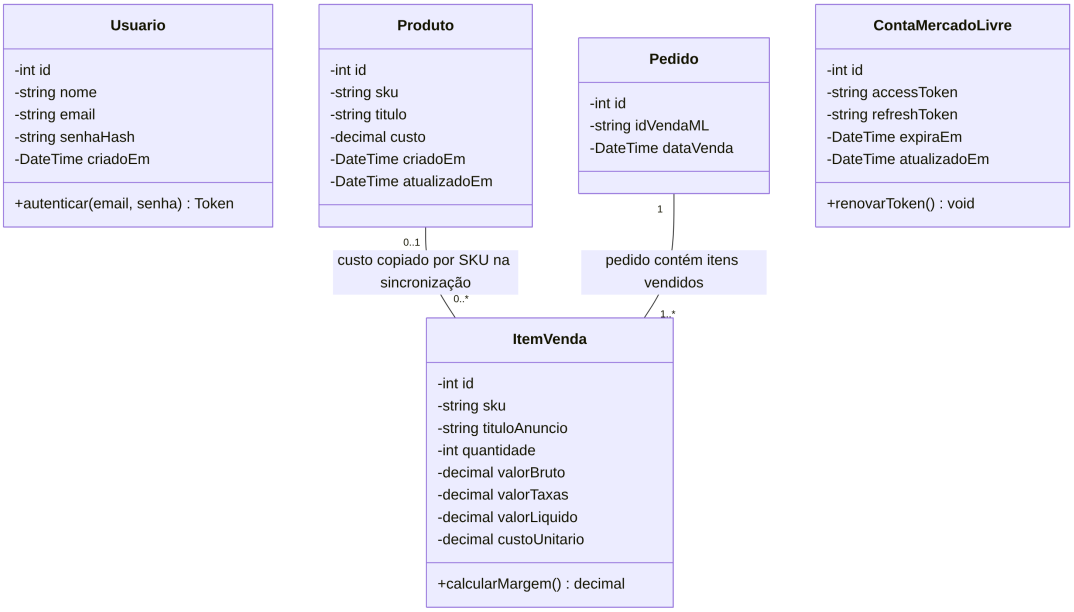
Figura 1 – Diagrama de casos de uso do sistema



Fonte: Autoria própria (2026).

A Figura 2 apresenta o diagrama de classes do sistema, correspondente ao modelo de dados previsto no esquema do ORM Prisma. As classes *Produto* e *ItemVenda* relacionam-se por meio do SKU: no momento da sincronização, cada item de venda obtém o custo do produto de mesmo SKU cadastrado no sistema e o copia para o próprio atributo *custoUnitario* do item, em vez de referenciá-lo indiretamente. Dessa forma, o cálculo da margem de cada venda é feito a partir de atributos armazenados no próprio *ItemVenda* (valor líquido e custo unitário), e não a partir do custo atual de *Produto*, garantindo que alterações posteriores no custo cadastrado não alterem retroativamente a margem de vendas já sincronizadas (RN03). A classe *Usuario* representa o gestor responsável por autenticar-se no sistema. A classe *ContaMercadoLivre* armazena as credenciais de acesso à API do Mercado Livre (token de acesso e *refresh token*), conforme RF03. Já a classe *Pedido* representa cada venda sincronizada do Mercado Livre, podendo conter um ou mais itens vendidos (RN05), relacionando-se com *ItemVenda* por meio de uma associação simples de multiplicidade *1* para *1..**. A multiplicidade *0..1* entre *Produto* e *ItemVenda* reflete a possibilidade de o produto vendido ainda não possuir custo cadastrado no momento da venda (RN02). Por fim, todos os atributos das classes são definidos como privados (-), sendo o acesso a eles mediado exclusivamente pelos métodos públicos (+) expostos por cada classe, em conformidade com o princípio de encapsulamento da orientação a objetos.

Figura 2 – Diagrama de classes do sistema



Fonte: Autoria própria (2026).

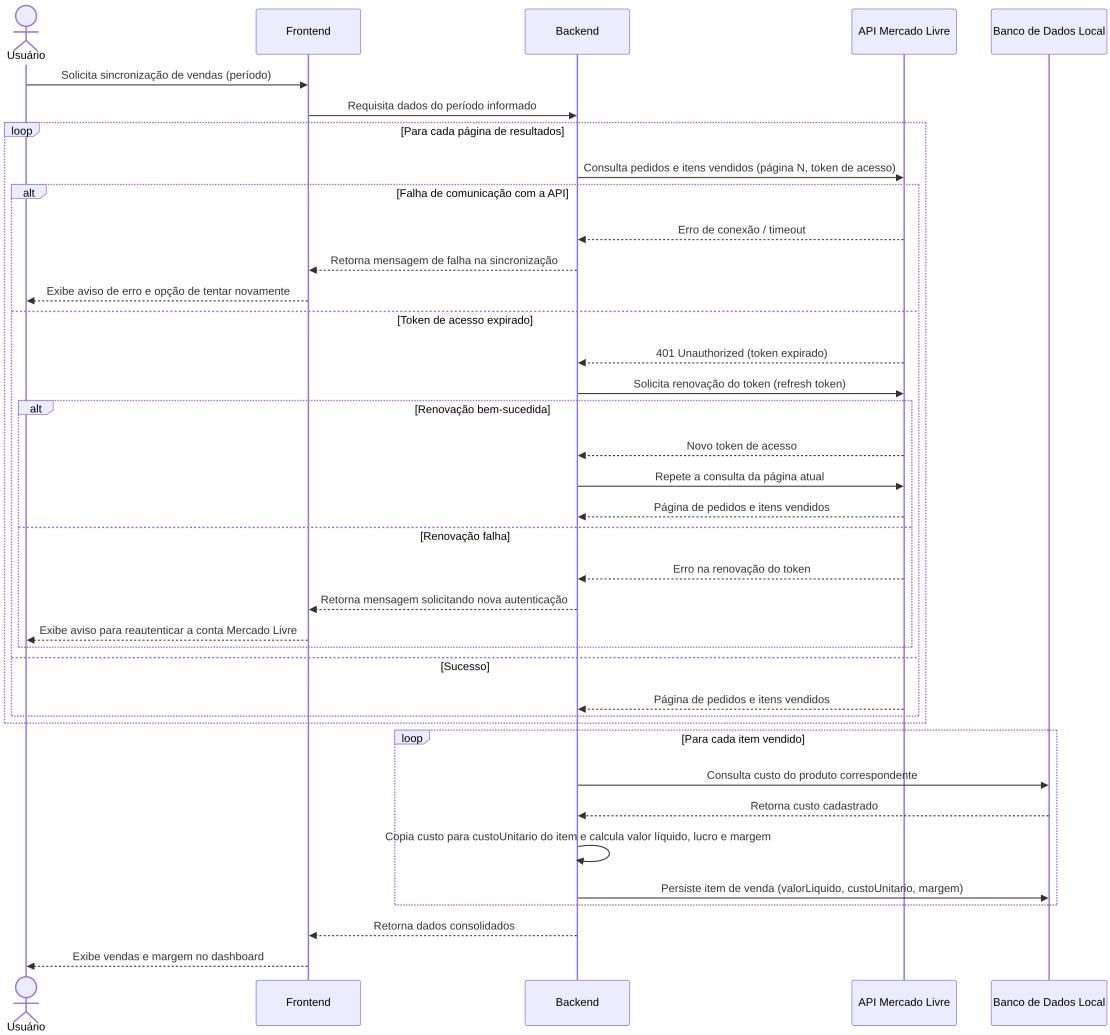
4.3.1 Diagramas de Sequência

Os diagramas de sequência apresentados nesta seção detalham os dois principais fluxos de integração do sistema com APIs externas: a sincronização das vendas do Mercado Livre

e o cálculo automático da margem de lucro, e a sincronização do custo dos produtos a partir do Tiny ERP.

A Figura 3 apresenta o fluxo de sincronização das vendas. Ao solicitar a sincronização, o *front-end* requisita ao *back-end* os dados do período informado; o *back-end*, por sua vez, consulta a API do Mercado Livre, obtém a lista de pedidos e itens vendidos e, para cada item, consulta o custo do produto correspondente no banco de dados local, copiando-o para o atributo *custoUnitario* do próprio item de venda antes de calcular o valor líquido, o lucro e a margem; o item de venda é então persistido — incluindo o *custoUnitario* fixado —, e o resultado é retornado ao *front-end* para exibição no *dashboard*.

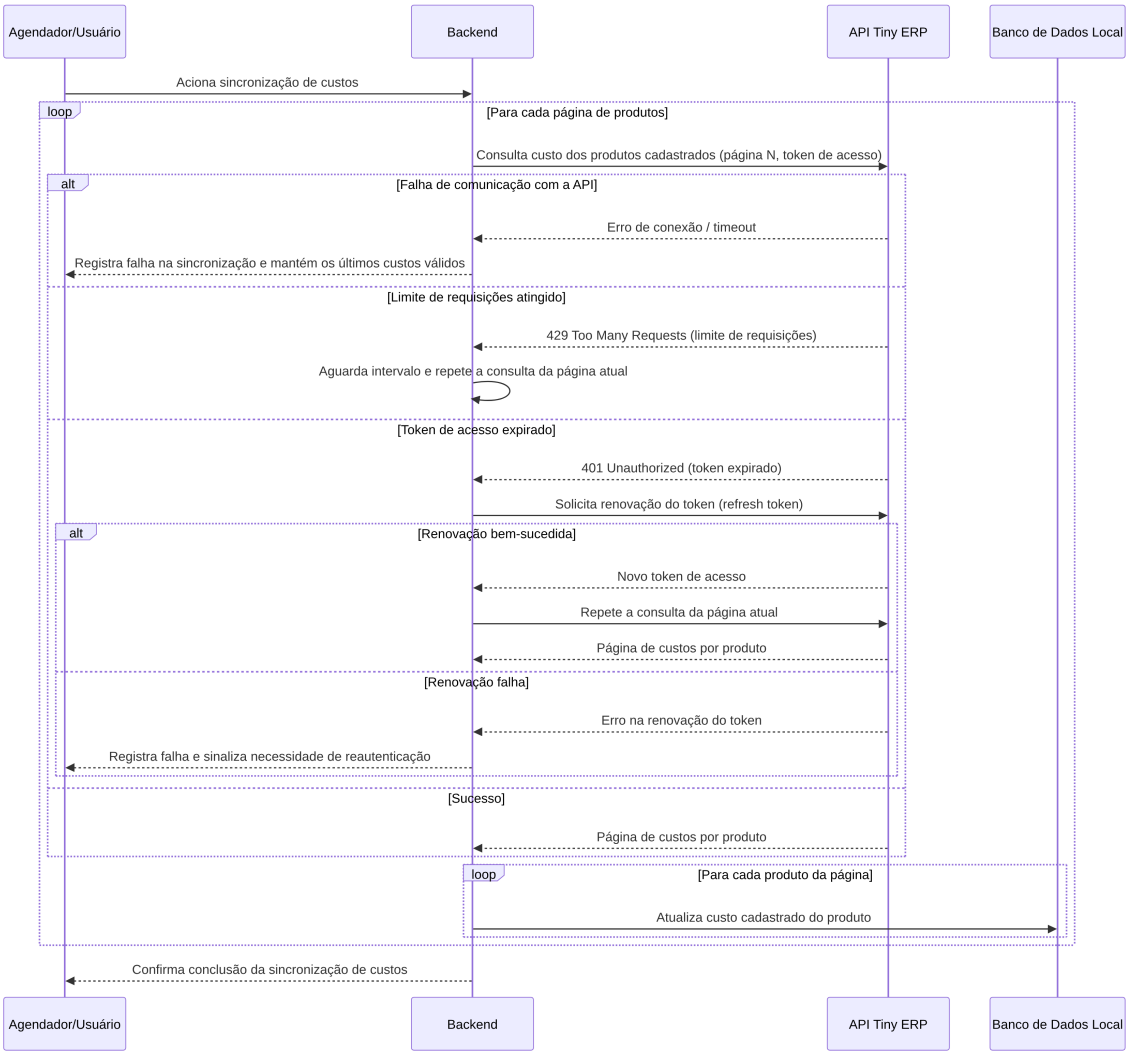
Figura 3 – Diagrama de sequência: sincronização de vendas do Mercado Livre e cálculo de margem



Fonte: Autoria própria (2026).

A Figura 4 apresenta o fluxo de sincronização de custos via Tiny ERP, utilizado para manter atualizado o custo dos produtos cadastrados sem a necessidade de preenchimento manual item a item.

Figura 4 – Diagrama de sequência: sincronização de custos via Tiny ERP



Fonte: Autoria própria (2026).

4.4 Sistema em Produção: Telas de Referência

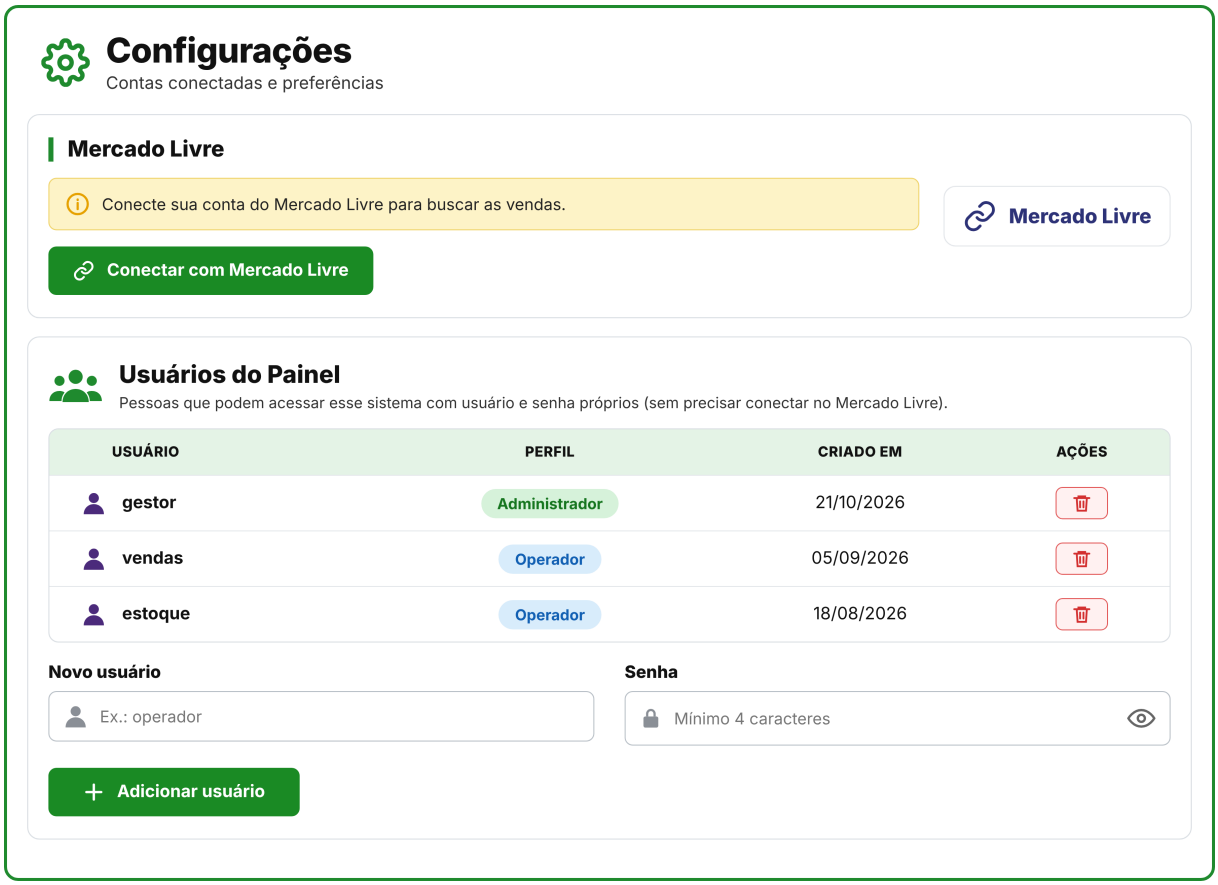
Esta seção apresenta as principais telas do sistema atualmente em produção na empresa, organizadas por área funcional. Esse sistema serviu de base para o levantamento dos requisitos apresentados na seção 4.2 e será reconstruído com a arquitetura e as tecnologias descritas no Capítulo 3.

4.4.1 Autenticação e Configurações

A Figura 5 apresenta a tela de configurações do sistema atualmente em produção, por meio da qual o usuário conecta a conta do Mercado Livre via OAuth 2.0 e gerencia os usuários

autorizados a acessar o painel. Essa funcionalidade deu origem aos requisitos RF01, RF02 e RF03, que o sistema proposto deverá contemplar.

Figura 5 – Tela de configurações: conexão com o Mercado Livre e gestão de usuários do painel

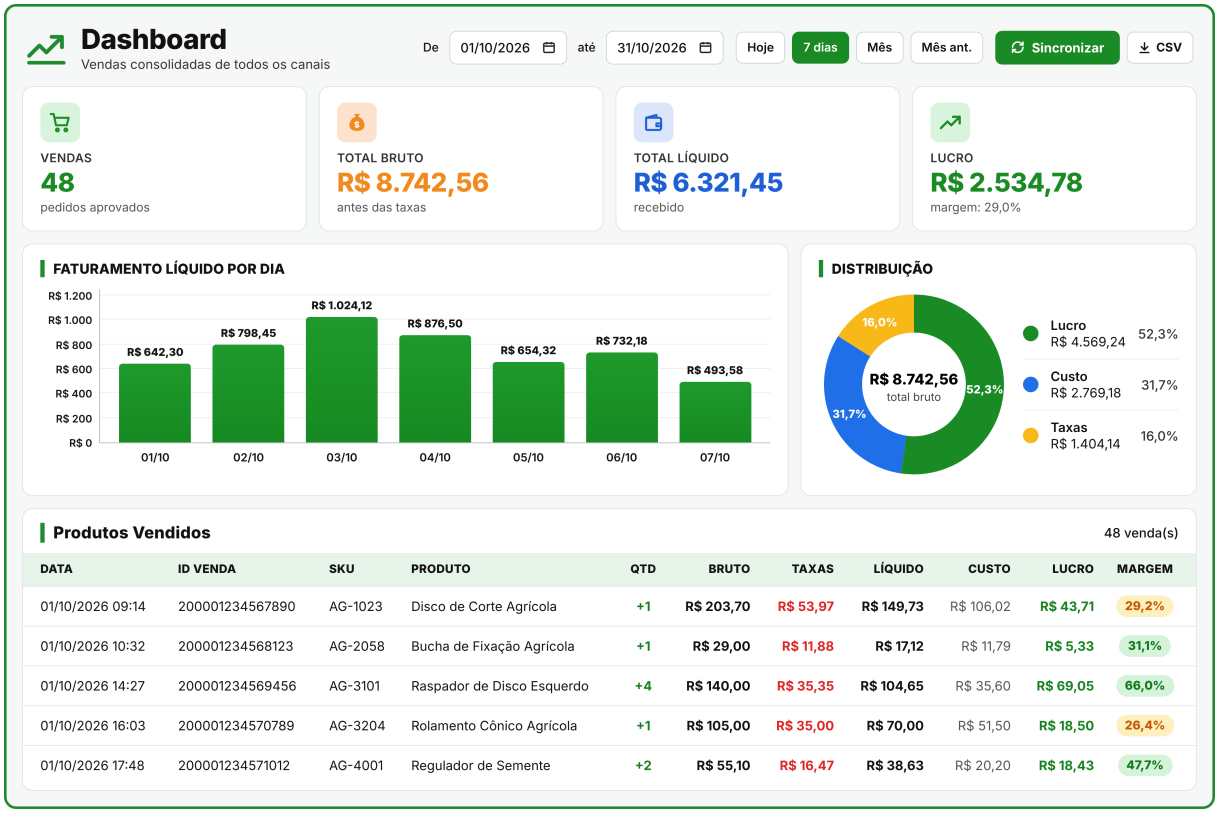


Fonte: Autoria própria (2026).

4.4.2 Dashboard e Indicadores de Vendas

A Figura 6 apresenta o dashboard principal do sistema atualmente em produção, com os indicadores consolidados de vendas, valor bruto, valor líquido, lucro e margem do período selecionado, os atalhos de filtro de período, o botão de sincronização com o Mercado Livre, o gráfico de faturamento líquido por dia e o gráfico de distribuição entre lucro, custo e taxas. Essas funcionalidades deram origem aos requisitos RF04, RF05, RF06, RF07, RF08 e RF09, que o sistema proposto deverá contemplar.

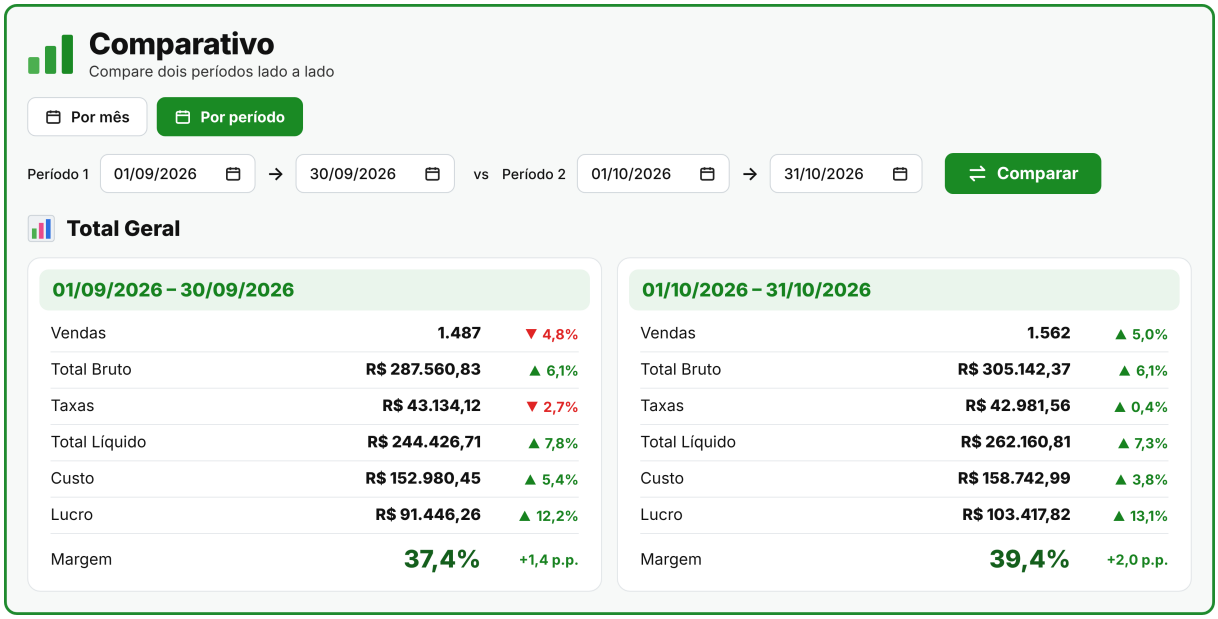
Figura 6 – Tela de dashboard: indicadores consolidados de vendas



Fonte: Autoria própria (2026).

Complementarmente, a Figura 7 apresenta a tela de comparativo do sistema atual, na qual o usuário pode comparar os indicadores de dois períodos distintos, lado a lado, com a variação percentual de cada indicador entre os dois períodos. Essa funcionalidade deu origem ao requisito RF10, que o sistema proposto deverá contemplar.

Figura 7 – Tela de comparativo entre dois períodos

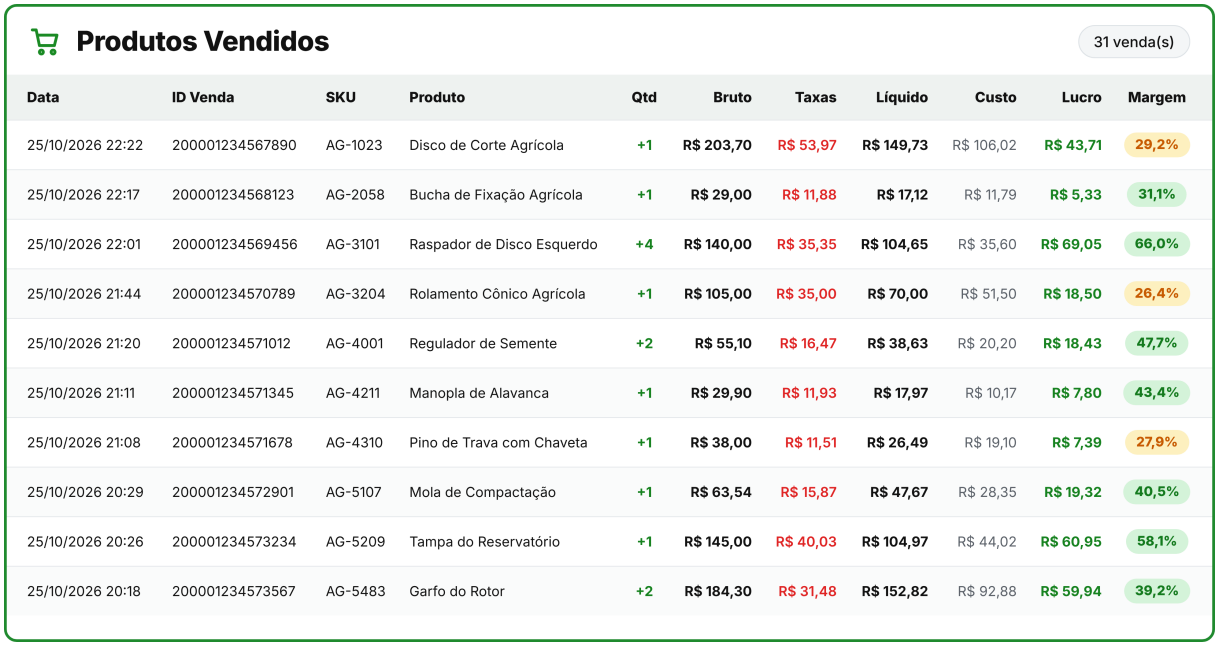


Fonte: Autoria própria (2026).

4.4.3 Listagem de Vendas do Mercado Livre

A Figura 8 apresenta a tela de listagem de vendas do sistema atualmente em produção, na qual cada linha corresponde a um item vendido, com data, canal, identificador do pedido, SKU, produto, quantidade e os valores de bruto, taxas, líquido, custo, lucro e margem. Quando o produto vendido ainda não possui custo cadastrado, o sistema considera o custo como zero para fins de cálculo: o lucro exibido iguala-se ao valor líquido e a margem é apresentada como 100%, uma limitação conhecida do cálculo automático. Essas funcionalidades deram origem aos requisitos RF11, RF12 e RF13, que o sistema proposto deverá contemplar.

Figura 8 – Tela de listagem de vendas do Mercado Livre



Produtos Vendidos — 31 venda(s)

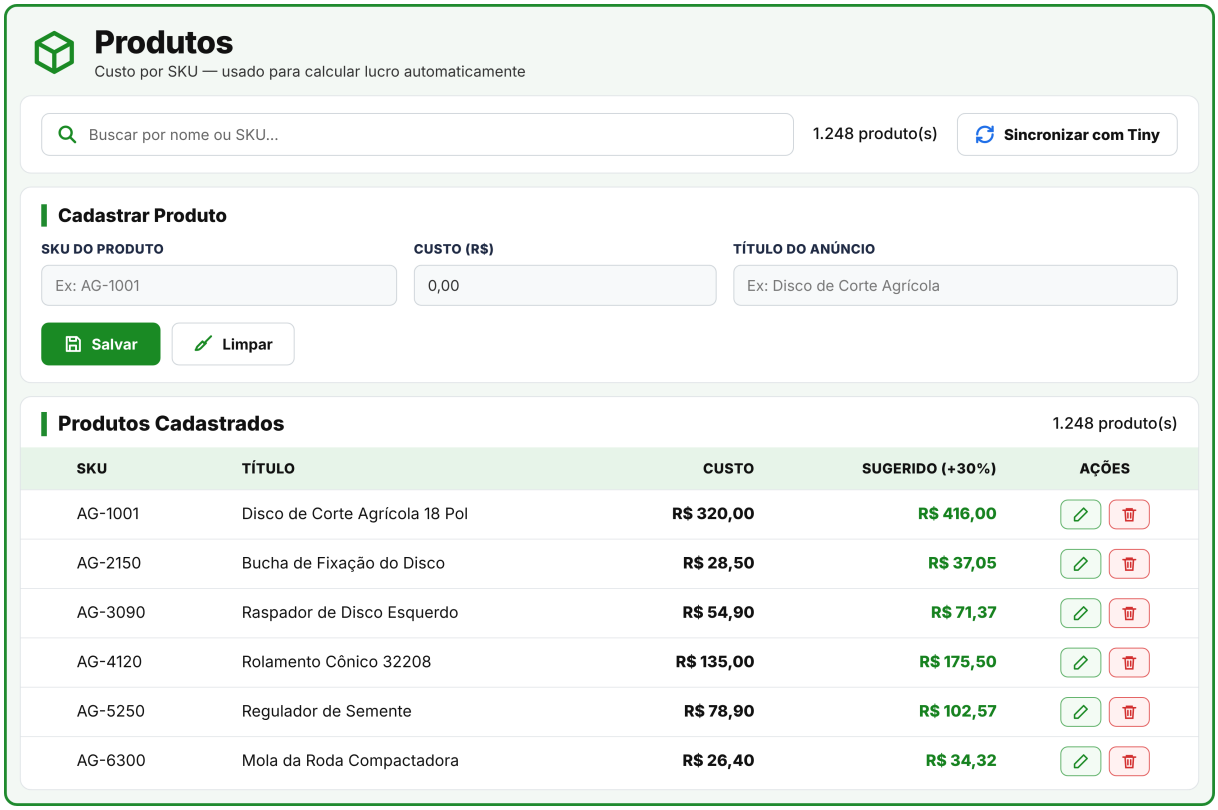
Data	ID Venda	SKU	Produto	Qtd	Bruto	Taxas	Líquido	Custo	Lucro	Margem
25/10/2026 22:22	200001234567890	AG-1023	Disco de Corte Agrícola	+1	R$ 203,70	R$ 53,97	R$ 149,73	R$ 106,02	R$ 43,71	29,2%
25/10/2026 22:17	200001234568123	AG-2058	Bucha de Fixação Agrícola	+1	R$ 29,00	R$ 11,88	R$ 17,12	R$ 11,79	R$ 5,33	31,1%
25/10/2026 22:01	200001234569456	AG-3101	Raspador de Disco Esquerdo	+4	R$ 140,00	R$ 35,35	R$ 104,65	R$ 35,60	R$ 69,05	66,0%
25/10/2026 21:44	200001234570789	AG-3204	Rolamento Cônico Agrícola	+1	R$ 105,00	R$ 35,00	R$ 70,00	R$ 51,50	R$ 18,50	26,4%
25/10/2026 21:20	200001234571012	AG-4001	Regulador de Semente	+2	R$ 55,10	R$ 16,47	R$ 38,63	R$ 20,20	R$ 18,43	47,7%
25/10/2026 21:11	200001234571345	AG-4211	Manopla de Alavanca	+1	R$ 29,90	R$ 11,93	R$ 17,97	R$ 10,17	R$ 7,80	43,4%
25/10/2026 21:08	200001234571678	AG-4310	Pino de Trava com Chaveta	+1	R$ 38,00	R$ 11,51	R$ 26,49	R$ 19,10	R$ 7,39	27,9%
25/10/2026 20:29	200001234572901	AG-5107	Mola de Compactação	+1	R$ 63,54	R$ 15,87	R$ 47,67	R$ 28,35	R$ 19,32	40,5%
25/10/2026 20:26	200001234573234	AG-5209	Tampa do Reservatório	+1	R$ 145,00	R$ 40,03	R$ 104,97	R$ 44,02	R$ 60,95	58,1%
25/10/2026 20:18	200001234573567	AG-5483	Garfo do Rotor	+2	R$ 184,30	R$ 31,48	R$ 152,82	R$ 92,88	R$ 59,94	39,2%

Fonte: Autoria própria (2026).

4.4.4 Produtos e Custos

Por fim, a Figura 9 apresenta a tela de produtos do sistema atual, na qual o usuário sincroniza automaticamente os produtos a partir da API do Tiny ERP, obtendo o SKU, o título e o custo de cada produto sem necessidade de preenchimento manual. Para os casos em que o SKU do produto no Mercado Livre diverge do SKU cadastrado no Tiny ERP, o usuário pode cadastrar manualmente o custo do produto, informando SKU, custo e título do anúncio. A tela também permite consultar a listagem de produtos já cadastrados, com o custo. Essas funcionalidades deram origem aos requisitos RF14, RF15 e RF16, que o sistema proposto deverá contemplar.

Figura 9 – Tela de produtos: cadastro e sincronização de custos



Fonte: Autoria própria (2026).

5 CONCLUSÃO

Este trabalho apresentou a proposta de uma aplicação web para o cálculo automático da margem de lucro das vendas realizadas pela Agrolan Peças Agrícolas no Mercado Livre, substituindo o controle atualmente realizado de forma manual em planilha eletrônica. A proposta busca automatizar a apuração do valor líquido, do lucro e da margem de cada venda, a partir da integração com a API do Mercado Livre e com a API do Tiny ERP, disponibilizando ao gestor um *dashboard* com indicadores consolidados que apoiem a tomada de decisão financeira da empresa.

A partir do referencial teórico, foi possível compreender que o controle manual de custos e margens é uma dificuldade recorrente entre micro e pequenas empresas, apontada na literatura como fator de risco à continuidade do negócio, e que o crescimento do comércio eletrônico e das comissões cobradas pelos marketplaces torna esse controle ainda mais suscetível a erros à medida que o volume de vendas aumenta. Também foi identificado que as soluções já existentes atuam de forma isolada: sistemas de gestão como Tiny e Bling cobrem a gestão fiscal e de estoque, mas não calculam a margem considerando as taxas específicas de cada marketplace, enquanto ferramentas de precificação, como a planilha disponibilizada pelo Sebrae, dependem do preenchimento manual de cada produto. Nesse sentido, a aplicação proposta busca integrar, em um único fluxo automatizado, os dados de vendas do marketplace e os dados de custo do ERP, calculando a margem de lucro sem a intervenção manual do usuário.

Na etapa de análise e projeto, foram definidos o escopo do sistema, os requisitos funcionais e não funcionais, as regras de negócio que orientam o cálculo da margem, os diagramas de casos de uso, de classes e de sequência, além das telas do sistema atualmente em produção na empresa, utilizadas como referência para o levantamento dos requisitos. A metodologia adotada, com apoio do método MoSCoW para priorização dos requisitos, contribuiu para organizar o escopo do sistema e delimitar as funcionalidades essenciais a serem desenvolvidas na primeira versão da aplicação. As tecnologias previstas, como NestJS, Prisma, SQLite, React, Vite e Mantine UI, foram escolhidas por atenderem às necessidades de desenvolvimento do *back-end* e do *front-end*, de persistência dos dados e de padronização da interface da aplicação, com autenticação via JWT e documentação da API por meio do Swagger.

Como principal contribuição, a proposta apresenta uma alternativa para tornar o cálculo da margem de lucro por venda mais ágil, confiável e escalável, substituindo um processo manual sujeito a erros e de difícil manutenção à medida que o volume de vendas da empresa cresce. Entre as limitações, destaca-se que o sistema ainda se encontra em fase de proposta, sem implementação completa ou validação com o uso real na empresa – etapa prevista para a continuidade deste trabalho. Além disso, o escopo definido abrange exclusivamente as vendas realizadas no Mercado Livre, não contemplando os demais canais de venda da Agrolan, como Shopee, Nuvemshop e loja física, tampouco as demais funcionalidades do sistema atualmente em produção, como emissão de etiquetas e cálculo de frete regional.

Dessa forma, considera-se que o objetivo da proposta foi alcançado, pois o trabalho definiu uma solução tecnológica viável para automatizar o cálculo da margem de lucro das vendas do Mercado Livre, delimitando um escopo adequado ao período de desenvolvimento previsto e estabelecendo as bases necessárias para a implementação do sistema na etapa seguinte deste trabalho.

REFERÊNCIAS

Bling. **Dashboard Margem por Pedido**. 2026a. Disponível em: https://ajuda.bling.com.br/hc/pt-br/articles/35658275718551-Dashboard-Margem-por-Pedido. Acesso em: 03 nov. 2026.

Bling. **Taxas do Mercado Livre: guia completo de custos e comissões**. 2026b. Disponível em: https://blog.bling.com.br/taxas-mercado-livre/. Acesso em: 13 set. 2026.

CARVALHO, V. S. d. *et al.* Gestão de custos e precificação: caso de estudo em uma empresa de pequeno porte de produção por encomenda. *In*: **Congresso Brasileiro de Custos**. Curitiba: [*s.n.*], 2008.

CLEGG, D.; BARKER, R. **Case Method Fast-Track: A RAD Approach**. Wokingham: Addison-Wesley, 1994.

DUBOIS, A.; KULPA, L.; SOUZA, L. E. d. **Gestão de Custos e Formação de Preços: Conceitos, Modelos e Ferramentas**. Rio de Janeiro: Atlas, 2019.

FIELDING, R. T. **Architectural Styles and the Design of Network-based Software Architectures**. 2000. Tese (Doutorado) — University of California, Irvine, 2000. Disponível em: https://ics.uci.edu/~fielding/pubs/dissertation/top.htm. Acesso em: 28 out. 2026.

GIL, A. C. **Como elaborar projetos de pesquisa**. 4. ed. São Paulo: Atlas, 2002.

GUEDES, G. T. A. **UML 2: Uma Abordagem Prática**. 3. ed. São Paulo: Novatec Editora, 2018.

HARDT, D. **The OAuth 2.0 Authorization Framework**. [*S.l.*], 2012. Disponível em: https://www.rfc-editor.org/rfc/rfc6749. Acesso em: 28 out. 2026.

JONES, M.; BRADLEY, J.; SAKIMURA, N. **JSON Web Token (JWT)**. [*S.l.*], 2015. Disponível em: https://www.rfc-editor.org/rfc/rfc7519. Acesso em: 28 out. 2026.

JUNIOR, A. R. M. S.; SANTOS, M. d. Arquitetura rest api e desenvolvimento de uma aplicação web service. **Projetos e Relatórios de Estágios**, 2019.

JUNIOR, S. S. B. *et al.* Gestão estratégica dos custos para formação do preço de venda: um estudo de caso em um supermercado de médio porte. *In*: **Congresso Internacional de Custos**. Florianópolis: [*s.n.*], 2005.

MAIA, P. A.; MELO, V. A.; DIAS, D. N. B. A importância da precificação para as micro e pequenas empresas: um estudo de caso numa empresa do ramo alimentício. **Revista Produção Online**, 2023.

Mercado Livre. **Tarifas e faturamento**. 2026. Disponível em: https://www.mercadolivre.com.br/ajuda/tarifas-e-faturamento_1472. Acesso em: 28 out. 2026.

Mercado Livre Developers. **API Docs**. 2026. Disponível em: https://developers.mercadolivre.com.br/pt_br/api-docs-pt-br-1. Acesso em: 13 set. 2026.

Mercado&Consumo. **E-commerce brasileiro movimenta R$ 125 bilhões e cresce 24,37% no 1º semestre**. 2026. Dados originalmente divulgados pela Abicomm. Disponível em: https://mercadoeconsumo.com.br/11/08/2026/ecommerce/e-commerce-brasileiro-movimenta-r-125-bilhoes-e-cresce-2437-no-1o-semestre/. Acesso em: 13 set. 2026.

Nubimetrics. **Centro de Aprendizagem**. 2026. Disponível em: https://online.nubimetrics.com/pt-br/centro-de-aprendizagem. Acesso em: 13 set. 2026.

PISSINATI, L. G.; DIAS, E. A margem de contribuição como métrica de precificação visando lucro para a empresa. **Revista Ibero-Americana de Humanidades, Ciências e Educação (REASE)**, 2022.

PURIFICAÇÃO, É. F. d. S. *et al.* Precificação no contexto das micro e pequenas empresas (mpes): podemos sofisticar ou ainda pecamos no básico? *In*: **Congresso USP de Controladoria e Contabilidade**. [*S.l.*: *s.n.*], 2020.

Sebrae. **Planilha e calculadora de precificação para marketplace**. 2026. Disponível em: https://sebrae.com.br/empreendedores/conteudos/gerenciar/planilha-e-calculadora-de-precificacao-para-marketplace. Acesso em: 13 set. 2026.

SILVA, F. A. d. S.; MESQUITA, R. F. d. **A contribuição do business intelligence na tomada de decisão nas micro e pequenas empresas**. 2024. Trabalho de Conclusão de Curso – Universidade Federal do Ceará, Campus Crateús, 2024. Disponível em: https://repositorio.ufc.br/handle/riufc/78446. Acesso em: 13 set. 2026.

Tiny ERP. **Documentação da API**. 2026. Disponível em: https://tiny.com.br/api-docs/api. Acesso em: 13 set. 2026.

TURBAN, E.; SHARDA, R.; DELEN, D. **Business Intelligence: A Managerial Approach**. Upper Saddle River, NJ: Pearson Prentice Hall, 2009.

Turbo Partners. **Tiny vs Bling: qual o melhor ERP para e-commerce?** 2026. Disponível em: https://blog.turbopartners.com.br/tiny-vs-bling-qual-o-melhor-erp-para-e-commerce/. Acesso em: 13 set. 2026.

ZAHAIKEVITCH, E. V. *et al.* Contabilidade gerencial no processo de tomada de decisão: o conhecimento dos empresários das micro e pequenas empresas. **Revista Produção Industrial & Serviços**, 2018.