



UNIVERSIDADE FEDERAL DE SANTA CATARINA  
CAMPUS ARARANGUÁ  
CENTRO DE CIÊNCIAS, TECNOLOGIAS E SAÚDE  
TECNOLOGIAS DA INFORMAÇÃO E COMUNICAÇÃO

Vitor Rodrigo Machado

DESENVOLVIMENTO DE UM SISTEMA DE GERENCIAMENTO DE PEDIDOS PARA  
MARKETPLACES

ARARANGUÁ

2023

Vitor Rodrigo Machado

DESENVOLVIMENTO DE UM SISTEMA DE GERENCIAMENTO DE PEDIDOS PARA  
MARKETPLACES

Trabalho de Conclusão de Curso submetido ao curso de Tecnologias da Informação e Comunicação do Centro de Ciências, Tecnologias e Saúde da Universidade Federal de Santa Catarina como requisito para a obtenção do título de Bacharel em Tecnologias da Informação e comunicação.

Orientador(a): Prof(a) Dra. Marina Carradore Sérgio

ARARANGUÁ

2023

Machado, Vitor Rodrigo  
DESENVOLVIMENTO DE UM SISTEMA DE GERENCIAMENTO DE PEDIDOS  
PARA MARKETPLACES / Vitor Rodrigo Machado ; orientadora, Marina  
Carradore Sérgio, 2023.  
50 p.

Trabalho de Conclusão de Curso (graduação) - Universidade  
Federal de Santa Catarina, Campus Araranguá, Graduação em  
Tecnologias da Informação e Comunicação, Araranguá, 2023.

Inclui referências.

1. Tecnologias da Informação e Comunicação. 2. Marketplace.  
3. Gerenciamento de Pedidos. I. Carradore Sérgio, Marina. II.  
Universidade Federal de Santa Catarina. Graduação em Tecnologias  
da Informação e Comunicação. III. Título.

Vitor Rodrigo Machado

**Desenvolvimento de um Sistema de Gerenciamento de Pedidos para Marketplaces**

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do título de “Bacharel em Tecnologias da Informação e Comunicação” e aprovado em sua forma final pelo Curso Tecnologias da Informação e Comunicação.

Araranguá, 11 de dezembro de 2023.

---

Fernando José Spanhol  
Coordenador do Curso

**Banca examinadora**

---

Prof.(a) Marina Carradore Sérgio, Dr.(a)  
Orientador(a)

---

Prof. Fabrício Herpich, Dr.  
Universidade Federal de Santa Catarina

---

Prof. Giovani Mendonça Lunardi, Dr.  
Universidade Federal de Santa Catarina

Araranguá, 2023

## **AGRADECIMENTOS**

Gostaria de expressar minha profunda gratidão a todos que me apoiaram durante a realização deste trabalho.

Agradeço especialmente à minha orientadora, Dra. Marina Carradore Sérgio, por sua paciência, sabedoria e orientações valiosas.

A minha família, pelo suporte incondicional e por acreditarem em mim.

Aos professores que contribuíram para minha formação, compartilhando seu conhecimento e inspiração, sou eternamente grato a todos que fizeram parte desta jornada.

## RESUMO

O comércio eletrônico tem crescido significativamente nos últimos anos, e os *marketplaces* se tornaram uma opção popular para vendedores e consumidores. No entanto, a gestão eficiente de pedidos em *marketplaces* é um desafio, envolvendo questões como confiança, prontidão e segurança. Um sistema de gerenciamento de pedidos eficiente é crucial para garantir a satisfação do cliente e a otimização dos procedimentos internos. Este trabalho de conclusão de curso aborda o desenvolvimento de um sistema de gerenciamento de pedidos para *marketplaces*, com o objetivo de atender às demandas específicas desse ambiente de comércio eletrônico. O sistema proposto visa incorporar funcionalidades essenciais, tais como registro detalhado de pedidos, integração com diferentes plataformas, notificações em tempo real, análise e relatórios de vendas, automação de processos, segurança e controle de acesso. A metodologia empregada incluiu o levantamento de requisitos por meio de entrevistas com vendedores e consumidores ativos em *marketplaces*, resultando em *insights* valiosos para a concepção do projeto. O sistema foi desenvolvido utilizando ferramentas e tecnologias como *MongoDB Atlas*, *Visual Studio Code*, *Node.js* e Selenium. Os resultados obtidos com o desenvolvimento do protótipo do sistema foram apresentados, destacando a visão geral do sistema e as ferramentas utilizadas. Conclui-se que o sistema desenvolvido atende às necessidades do mercado, proporcionando uma gestão de pedidos mais ágil, segura e adaptada às nuances do ambiente dinâmico dos *marketplaces*. Além disso, são apontadas possíveis direções para pesquisas futuras, refletindo a ênfase do curso na inovação e aplicação prática de tecnologias em diferentes contextos do *e-commerce* e gerenciamento de sistemas.

**Palavras-chave:** Marketplace; Sistema de Gerenciamento de Pedidos; Comércio Eletrônico.

## ABSTRACT

E-commerce has grown significantly in recent years, and marketplaces have become a popular option for sellers and consumers. However, efficient order management on marketplaces is a challenge, involving issues such as trust, promptness, and security. An efficient order management system is crucial for ensuring customer satisfaction and optimizing internal procedures. This term paper addresses the development of an order management system for marketplaces, with the aim of meeting the specific demands of this e-commerce environment. The proposed system aims to incorporate essential features such as detailed order registration, integration with different platforms, real-time notifications, sales analysis and reporting, process automation, security, and access control. The methodology employed included requirements gathering through interviews with sellers and consumers active in marketplaces, resulting in valuable insights for the design of the project. The system was developed using tools and technologies such as MongoDB Atlas, Visual Studio Code, Node.js and Selenium. The results obtained from the development of the system prototype were presented, highlighting the system overview and the tools used. The results obtained from the development of the system prototype were presented, highlighting the system overview and the tools used. The conclusion is that the system developed meets the needs of the market, providing more agile, secure order management adapted to the nuances of the dynamic environment of marketplaces. In addition, possible directions for future research are pointed out, reflecting the course's emphasis on innovation and practical application of technologies in different contexts of e-commerce and systems management.

**Keywords:** Marketplace; Order Management System; E-commerce.

## LISTA DE FIGURAS

Figura 1 - Diagrama do ciclo de vida iterativo e incremental .....	29
Figura 2 - Diagrama de caso de uso .....	31
Figura 3 - Página de login .....	34
Figura 4 - Página Inicial .....	35
Figura 5 - Página Gestão de Anúncios .....	36
Figura 6 - Seleção de marketplace .....	37
Figura 7 - Lista de produtos .....	37
Figura 8 - Página de Pedidos .....	38
Figura 9 - Página de dados .....	39
Figura 10 - Página de configuração.....	40
Figura 11 - Página de contato.....	41

## LISTA DE QUADROS

Quadro 1 - Requisitos Funcionais .....	31
Quadro 2 - Requisitos Não Funcionais.....	31

## LISTA DE ABREVIATURAS E SIGLAS

TIC	Tecnologias da Informação e Comunicação
CSS	<i>Cascading Style Sheets</i>
HTML	<i>HyperText Markup Language</i>
JS	Javascript
API	<i>Application Programing Interface</i>
REST	<i>Representational State Transfer</i>
WWW	<i>World Wide Web</i>
W3C	<i>World Wide Web Consortium</i>
URL	<i>Uniform Resource Locator</i>
HTTP	<i>Hypertext Transfer Protocol</i>
WHATWG	<i>Web Hypertext Application Technology Working Group</i>
NPM	<i>Node Package Manager</i> –
Server-side	Lado do servidor
Client-side	Lado do Cliente
JSON	<i>Java Script Object Notation</i>
OOP	Programação Orientada a Objetos
GUI	Interface gráfica do utilizador

## SUMÁRIO

<b>1 INTRODUÇÃO</b>	<b>13</b>
1.1 PROBLEMÁTICA	14
1.2 JUSTIFICATIVA	15
1.3 OBJETIVOS	16
<b>1.3.1 Objetivo Geral:</b>	<b>17</b>
<b>1.3.2 Objetivos Específicos:</b>	<b>17</b>
1.4 ADERÊNCIA AO CURSO DE TIC	17
1.5 ESTRUTURA DO TEXTO	18
<b>2 REFERENCIAL TEÓRICO</b>	<b>19</b>
2.1 INTERNET	19
2.2 MARKETPLACE	20
2.3 APLICAÇÃO WEB	21
<b>2.3.1 Hospedagem na web</b>	<b>21</b>
2.4 FRONT-END	22
<b>2.4.1 HTML</b>	<b>23</b>
<b>2.4.2 CSS</b>	<b>23</b>
<b>2.4.3 Javascript</b>	<b>23</b>
<b>2.4.4 BootStrap</b>	<b>24</b>
2.5 BACKEND	25
2.6 SELENIUM	25
2.7 BANCO DE DADOS	26
<b>2.7.1 MongoDB</b>	<b>27</b>
<b>2.7.2 MongoDB Atlas</b>	<b>27</b>
2.8 NODE.JS E NODE PACKAGE MANAGER (NPM)	28
2.9 VISUAL STUDIO CODE	28
<b>3 METODOLOGIA</b>	<b>29</b>
3.1 MODELO INTERATIVO E INCREMENTAL	29
3.2 LEVANTAMENTO E ANÁLISE DE REQUISITOS	30
<b>3.2.1 Requisitos funcionais</b>	<b>31</b>
<b>3.2.2 Requisitos não funcionais</b>	<b>31</b>
<b>4 RESULTADOS E DISCUSSÕES</b>	<b>32</b>

4.1 LEVANTAMENTO DE REQUISITOS .....	32
4.2 VISÃO GERAL DO SISTEMA.....	32
4.3 ANÁLISE DAS PÁGINAS FUNCIONAIS.....	33
<b>4.3.1 Login .....</b>	<b>33</b>
<b>4.3.2 Home .....</b>	<b>34</b>
<b>4.3.3 Gestão de Anúncios .....</b>	<b>35</b>
<b>4.3.4 Pedidos.....</b>	<b>38</b>
<b>4.3.5 Dados.....</b>	<b>39</b>
<b>4.3.6 Configurações.....</b>	<b>40</b>
<b>4.3.7 Contato .....</b>	<b>40</b>
4.4 FUNCIONALIDADES EM DESENVOLVIMENTO .....	41
4.5 ARMAZENAMENTO E APRESENTAÇÃO DE DADOS .....	42
4.6 CONSIDERAÇÕES FINAIS .....	43
<b>5 CONSIDERAÇÕES FINAIS.....</b>	<b>45</b>
<b>REFERÊNCIAS.....</b>	<b>47</b>

## 1 INTRODUÇÃO

O dinamismo crescente no mercado não apenas intensifica a concorrência e a incerteza nos setores econômicos, mas também apresenta novos desafios de gestão, demandando práticas inovadoras na concepção e administração das cadeias de suprimentos (Simão; Gonçalves, 2015). As empresas, cada vez mais dependentes da capacidade gerencial, buscam elevar sua competitividade, reduzir custos e aprimorar a qualidade em comparação com os concorrentes (Guissoni; Farinha, 2019). Nesse contexto, torna-se imperativo que as empresas desenvolvam novos modelos de gestão, visando atender e, se possível, superar as expectativas dos clientes (Durski, 2003).

Com o objetivo de aprimorar seu desempenho, empresas de diversos setores têm implementado novas estratégias logísticas. A excelência nessas atividades não apenas resulta em melhorias significativas no atendimento ao cliente, na eficiência e na confiabilidade dos processos logísticos, mas também contribui para a redução de custos (Leuschner; Charvet; Rogers, 2013).

As novas tecnologias de comunicação têm proporcionado facilidades que transformaram os relacionamentos no mundo dos negócios, principalmente entre empresas e consumidores. Nesse contexto, uma das estratégias em voga é a entrada no mercado virtual.

Logo, é notável que no século XXI, os avanços tecnológicos exponenciais foram impulsionados pela disseminação generalizada do acesso à internet. Nesse cenário, as operações comerciais e o comércio eletrônico desempenharam papéis fundamentais, contribuindo de maneira decisiva para transformações organizacionais (Franco; Maia; Souza Filho, 2020).

No Brasil, em 2017, o faturamento do *e-commerce* atingiu aproximadamente 60 bilhões de reais (Ebit, 2018). Assim como no mundo físico, esse mercado abriga grandes empresas, geralmente líderes em seus segmentos, com uma significativa participação nas vendas. Além delas, há a presença de pequenas e médias empresas engajadas em vendas online. Esse modelo, conhecido como *marketplace*, é um ambiente virtual onde compradores e vendedores se encontram para realizar transações comerciais. Esse mercado desempenha três funções principais: facilitar o encontro entre compradores e vendedores, simplificar a troca de informações, bens, serviços e pagamentos relacionados à transação, e oferecer uma infraestrutura que viabilize o funcionamento eficiente desse ambiente (Cheah; Phau; Liang, 2015; Turban; Lee; King; Liang; Turban, 2009).

Em um mundo cada vez mais conectado e digital, os *marketplaces* surgem como plataformas essenciais para os negócios online. Segundo Pasquali (2023), o comércio eletrônico tem apresentado crescimento contínuo nos últimos anos, principalmente após as restrições causadas pela pandemia do Covid-19. O período de pandemia ocasionou um aumento significativo das vendas através de *marketplaces*. Contudo, gerir pedidos em diversos *marketplaces* pode ser uma tarefa complexa e trabalhosa. Considerando essas necessidades, é imperativo ter sistemas eficazes de gerenciamento de pedidos.

Lovelock, Wirtz e Hemzo (2011, p. 143) afirmam que a utilização de vários canais para entregar o mesmo serviço não tem somente várias implicações de custo para a empresa fornecedora, mas também afetam em muito a natureza da experiência do serviço para o cliente pelas diferentes velocidades, ambientes, interação, e graus de praticidade oferecidos. Gerando uma necessidade para garantir a satisfação do cliente, que depende da eficiência com que suas ordens são processadas e entregues.

Os vendedores, por sua vez, estão sob constante pressão para otimizar seus processos. Em um ambiente altamente competitivo como o dos *marketplaces*, pequenos deslizes podem resultar em perdas significativas. Um sistema eficiente não só garante a satisfação do cliente, mas também proporciona aos vendedores uma vantagem competitiva, facilitando a gestão de pedidos de múltiplas fontes com precisão. Tendo em vista estes desafios, a implementação de um sistema capaz de auxiliar e gerenciar essas tarefas se torna essencial. Esse sistema deve ser capaz de consolidar pedidos de diferentes plataformas, garantir atualizações em tempo real e permitir a automação de processos. Isso não apenas melhora a eficiência operacional, mas também ajuda os vendedores a se adaptarem rapidamente às mudanças no comportamento do consumidor ou às tendências do mercado.

## 1.1 PROBLEMÁTICA

Conforme afirma Noronha et al. (2022), o comércio eletrônico experimenta um crescimento notável, com vendas online aumentando significativamente de cerca de \$25 bilhões no início do século XXI para mais de \$175 bilhões em 2010. Este crescimento exponencial sublinha a necessidade crítica de sistemas avançados de gestão de inventário e pedidos para lidar com o aumento da demanda e complexidade nas operações de *e-commerce*. No entanto, a

transição para o ambiente virtual impõe às empresas desafios logísticos substanciais, especialmente no que diz respeito à gestão de inventário e pedidos.

Segundo os autores, os componentes críticos para o sucesso do *e-commerce* incluem a gestão eficiente do inventário e a capacidade de rastrear a disponibilidade de estoque, elementos fundamentais para melhorar a experiência do cliente e lidar com o aumento de pedidos em um ambiente digital em constante mudança.

Diante dos desafios enfrentados por empresas de *e-commerce*, que incluem a ausência de um controle de acesso eficaz ao inventário, a incapacidade de rastrear mudanças e atividades em tempo real, e a falta de segurança nos dados armazenados, torna-se necessário analisar como essas empresas superam tais obstáculos. Conforme os autores Mohamed et al. (2021), os desafios no *e-commerce* envolvem questões como confiança, prontidão e segurança, que são fundamentais para a gestão eficiente de inventário e pedidos. A investigação de como as empresas de *e-commerce* no Brasil enfrentam e superam esses desafios logísticos, identificando melhores práticas e soluções eficazes, é essencial para assegurar operações logísticas eficientes e seguras.

## 1.2 JUSTIFICATIVA

Um sistema de gerenciamento eficiente revela-se essencial no contexto dos *marketplaces*, proporcionando uma base sólida para monitorar, organizar e executar pedidos de clientes de maneira precisa e oportuna. A garantia da entrega de produtos dentro dos prazos estabelecidos é essencial não apenas para a satisfação do cliente, mas também para a otimização dos procedimentos internos, minimizando falhas e ampliando a produtividade operacional. De acordo com a TrueCommerce (2023), um sistema eficaz de gerenciamento de estoque pode reduzir custos associados ao armazenamento e ao desperdício, além de melhorar a precisão no gerenciamento de estoque, o que é vital para a experiência de compra do cliente.

Com base na experiência acumulada no ambiente digital e na gestão de pequenos negócios em *marketplaces*, torna-se evidente a importância crucial de aprimorar a experiência de compra e venda nestes espaços virtuais. Essa realidade sublinha a necessidade imperativa de desenvolver sistemas de gerenciamento de pedidos eficientes. Estes sistemas devem não apenas responder às exigências do mercado, mas também adaptar-se às complexidades inerentes ao *e-commerce*, que envolve uma ampla variedade de lojas e plataformas.

Neste contexto, a Shopify (2023) ressalta a eficácia desses sistemas na gestão de produtos com baixo volume de vendas. Tal gestão é fundamental para otimizar o uso do espaço de armazenamento e para sincronizar os dados de vendas e estoque através de múltiplos canais. Isso, por sua vez, contribui significativamente para a eficiência operacional, garantindo que os recursos sejam utilizados da maneira mais produtiva possível e melhorando a experiência geral tanto para os vendedores quanto para os consumidores em marketplaces.

Para atender eficazmente às demandas específicas de um *marketplace*, o sistema proposto deve incorporar funcionalidades essenciais:

- Registro Detalhado de Pedidos: Captura de informações cruciais, como dados do cliente, produtos adquiridos, quantidade, preço e prazo de entrega;
- Integração com Diferentes Plataformas: Compatibilidade com diversos *marketplaces*, consolidando os pedidos de diferentes plataformas em uma única interface;
- Notificações em Tempo Real: Alertas imediatos sobre novos pedidos, alterações ou cancelamentos, garantindo a agilidade nas respostas;
- Análise e Relatórios de Vendas: Recursos que capacitam os vendedores a compreender suas vendas, identificar tendências e áreas de melhoria;
- Automação de Processos: Redução da intervenção manual, assegurando rapidez nas operações e minimizando o risco de erros;
- Segurança e Controle de Acesso: Restrição do acesso às informações de pedidos apenas a funcionários autorizados, assegurando a confidencialidade dos dados dos clientes.

Ao atender a essas funcionalidades, o sistema não apenas se alinha às demandas do mercado, mas também contribui significativamente para a eficiência operacional, proporcionando uma gestão de pedidos mais ágil, segura e adaptada às nuances do ambiente dinâmico dos *marketplaces*.

### 1.3 OBJETIVOS

Para melhor entendimento do trabalho apresentado, seus objetivos foram divididos entre objetivos gerais e específicos.

### 1.3.1 Objetivo Geral:

Desenvolver um sistema web para o gerenciamento eficiente de pedidos em diferentes marketplaces, aprimorando a operacionalização e experiência de compra e venda.

### 1.3.2 Objetivos Específicos:

- Aprimorar a administração de pedidos em lojas virtuais através de uma interface intuitiva e funcional;
- Integrar com diferentes plataformas de *marketplace* para centralizar a gestão de vendas;
- Implementar um sistema de registro detalhado e automatizado de pedidos, otimizando o processo de gestão e análise.

## 1.4 ADERÊNCIA AO CURSO DE TIC

O desenvolvimento de um sistema de gerenciamento de pedidos para marketplaces, realizado neste projeto, ressoa com a missão do Curso de Tecnologias da Informação e Comunicação (TIC) da UFSC. O curso, prepara os estudantes para enfrentar desafios tecnológicos contemporâneos, o que se encaixa na criação de soluções eficazes em ambientes de *e-commerce*. Este trabalho, incorporando conceitos e tecnologias serve como um exemplo da aplicação prática dos conhecimentos adquiridos no curso. A realização deste projeto, juntamente com outros TCCs semelhantes, destaca o papel vital da UFSC Araranguá na formação de profissionais qualificados, capazes de contribuir significativamente para o setor de tecnologia.

Este trabalho aborda temas semelhantes a outros apresentados no curso, como o de Padilha e Farias (2017) sobre '*E-commerce e marketing* digital no desenvolvimento de um site de recomendações: o caso iobras', e o trabalho de William Sebastião Teixeira (2023), focado no 'Desenvolvimento de um sistema para o controle de estoque de uma Instituição de Segurança Pública do Estado de Santa Catarina'. Ambos os trabalhos, embora distintos em aplicação, refletem a ênfase do curso na inovação e aplicação prática de tecnologias em diferentes contextos do *e-commerce* e gerenciamento de sistemas.

## 1.5 ESTRUTURA DO TEXTO

Este documento está organizado em cinco capítulos para uma compreensão clara do tema:

- Capítulo um traz a introdução, apresentando o contexto dos *marketplaces* e a importância da gestão de pedidos;
- Capítulo dois discute o referencial teórico, abordando as tendências e desafios dos *marketplaces* e a relevância de sistemas de gestão;
- No capítulo três, é detalhada a metodologia empregada, incluindo o levantamento de requisitos e as tecnologias selecionadas;
- O capítulo quatro apresenta os resultados do desenvolvimento, ilustrando as funcionalidades e interfaces do sistema proposto;
- Por fim, o quinto capítulo oferece uma reflexão sobre o trabalho realizado e possíveis direções para pesquisas futuras.

Com essa estrutura, o leitor poderá compreender a importância, o processo e os resultados do desenvolvimento do sistema de gerenciamento de pedidos para *marketplaces*.

## 2 REFERENCIAL TEÓRICO

Neste capítulo será realizada a fundamentação teórica sobre os principais temas envolvendo o trabalho.

### 2.1 INTERNET

O advento da internet remonta à época da Guerra Fria, quando os Estados Unidos perceberam a necessidade de estabelecer um meio de comunicação remota para facilitar o compartilhamento de informações durante o período bélico. Os estudos foram conduzidos pelo Departamento de Defesa dos Estados Unidos, notadamente pela *Advanced Research Projects Agency* (ARPA). O marco inaugural desse empreendimento ocorreu em 29 de outubro de 1969, com a estabelecimento da primeira conexão entre a Universidade da Califórnia e o Instituto de Pesquisa de Stanford, delineando assim o protótipo da primeira rede de internet conhecida como Arpanet (CASTELL, 2003).

A evolução subsequente da internet ao longo dos anos culminou na formação de um sistema global, acessível por meio de uma diversidade de dispositivos e caracterizado por uma facilidade e rapidez de acesso, com os sistemas operando em servidores. Essa infraestrutura refinada, facilita sobremaneira as atividades dos usuários. A ubiquidade da internet transformou-a em uma ferramenta instrumental no desenvolvimento de programas contemporâneos destinados ao gerenciamento eficiente de diversas operações, conforme apresentado por Cesare Pautasso (2023).

A convergência da acessibilidade global, a diversidade de dispositivos e a centralização operacional em servidores consolida a internet como uma plataforma ideal para a concepção e execução de programas de gestão, proporcionando aos usuários a capacidade de acesso a informações essenciais de maneira remota e dinâmica, independentemente da localização física ou do dispositivo disponível (SOMMERVILLE, 2011).

## 2.2 MARKETPLACE

Conforme descrito por Becker (2007), os *marketplaces* representam uma evolução significativa do comércio eletrônico, cuja origem remonta à década de 70 e 80 nos Estados Unidos, com a chegada do *Electronic Data Interchange* (EDI). Essa tecnologia foi fundamental para permitir a troca de informações empresariais de forma eletrônica. Empresas começaram a utilizar o EDI para enviar pedidos e realizar transferências eletrônicas através de computadores. Esse processo marcou o início da primeira fase do comércio eletrônico.

Esses ambientes virtuais operam como extensas praças de comércio online, assemelhando-se a grandes shoppings virtuais, nos quais os consumidores podem acessar uma ampla variedade de produtos provenientes de distintos vendedores. Essa estrutura estabelece uma parceria funcional em que as plataformas oferecem aos comerciantes a infraestrutura técnica necessária para administrar suas lojas virtuais, enquanto os comerciantes cooperam com seus produtos, contribuindo para a relevância contínua da plataforma (HUBSPOT, 2023).

O crescimento exponencial dessas plataformas tem sido notável nos últimos anos, testemunhando a proliferação de inúmeras lojas que optam por operar de forma virtual, alimentando um ciclo virtuoso que gera mais oportunidades para os comerciantes anunciarem seus produtos, ao passo que proporciona aos consumidores uma gama cada vez mais ampla de opções. Este fenômeno reflete uma tendência marcante na transição para o comércio online, consolidando os *marketplaces* como facilitadores cruciais nesse novo paradigma de negócios (TEIXEIRA, 2015).

Segundo Neto (2003), os marketplaces são plataformas vitais no cenário atual do comércio eletrônico, oferecendo um vasto campo para vendedores e consumidores. Entre os principais marketplaces do mercado, destacam-se:

- Mercado Livre®: Operando em diversos países, é um dos maiores da América Latina, permitindo que usuários anunciem, comprem e vendam produtos e serviços online, além de personalizar suas lojas virtuais.
- Amazon®: Globalmente reconhecida, a Amazon® no Brasil tem crescido em popularidade, oferecendo uma ampla gama de produtos e serviços, incluindo o *Amazon Prime* e *Amazon Web Services*.
- AliExpress®: Gigante de marketplace chinesa, se destaca no cenário do comércio eletrônico por abrir oportunidades para vendedores brasileiros desde 2021, com

baixas taxas de comissão e entrega competitiva, é uma das maiores plataformas de marketplace do Brasil,

- Magazine Luiza® (Magalu): Com uma forte presença online, o Magalu® possui um dos marketplaces mais acessados do Brasil, integrando lojas físicas e digitais.
- Elo7®: Focado no nicho de artesanato, oferecendo produtos autorais e artesanais.

## 2.3 APLICAÇÃO WEB

Conforme destacado por Noletto (2020), as aplicações web são programas que podem ser executados diretamente em um navegador, sendo distribuídos de maneira ampla para qualquer pessoa com uma conexão ativa à internet, dispensando a necessidade de instalação no dispositivo do usuário.

Essas aplicações apresentam elevado grau de adaptabilidade, possibilitando a criação de diversos tipos de funcionalidades, como edição de imagens, gerenciamento de e-mails, chats, compra e venda de itens, além da emissão de ingressos.

Essas aplicações também incluem servidores web, responsáveis por processar os dados dos usuários. Conforme informações disponíveis no site da Mozilla (2023), os servidores web operam utilizando o protocolo HTTP, que constitui a base da *World Wide Web* para a transferência de dados entre servidores e clientes. As aplicações são hospedadas nos servidores web, os quais desempenham o papel chave de processar as solicitações recebidas dos clientes e enviar as respostas correspondentes.

### 2.3.1 Hospedagem na web

A hospedagem web é um serviço essencial que viabiliza a presença de sites na internet, possibilitando que tanto indivíduos quanto organizações aloquem seus sites em servidores através do aluguel de espaço físico de empresas provedoras desse serviço. Nesse cenário, todos os dados e arquivos indispensáveis para o funcionamento do site são armazenados e gerenciados pelo servidor da empresa hospedeira. A responsabilidade dessa empresa engloba a manutenção contínua do servidor, a implementação de medidas de segurança no ambiente de hospedagem e a garantia da transferência bem-sucedida dos dados, como textos, fotos e outros arquivos, para os navegadores dos usuários (Buratto, 2023).

Adicionalmente, a hospedagem web pode ser realizada localmente, utilizando o próprio computador do usuário como servidor ativo. Essa abordagem reduz a barreira de entrada devido à menor despesa associada em comparação ao aluguel de um servidor externo. No entanto, é importante destacar que essa opção pode apresentar desafios, especialmente em relação ao escalonamento e crescimento sustentável do site ao longo do tempo, principalmente se a quantidade de solicitações ao servidor aumentar significativamente (Hostinger, 2023).

Essa dualidade de opções de hospedagem proporciona flexibilidade aos usuários, permitindo a escolha da abordagem mais adequada às necessidades e recursos específicos de cada projeto online. Cada modelo possui suas vantagens e desafios, e a decisão entre hospedagem externa e local deve ser cuidadosamente ponderada, considerando as projeções de tráfego, as exigências de escalabilidade e os recursos financeiros disponíveis.

## 2.4 FRONT-END

O desenvolvimento *front-end*, conforme enfatizado por Machado (2022), representa a porção de um website com a qual o usuário final interage diretamente. Este domínio abrange todos os elementos visíveis e interativos de uma página web, englobando a criação e implementação de interfaces visuais, elementos de design, e todas as funcionalidades acessíveis diretamente pelo usuário. Em termos mais simples, o *front-end* constitui a camada de um website que os usuários visualizam e interagem, formando a interface durante a navegação.

A influência significativa do desenvolvimento *front-end* na experiência do usuário online não se restringe apenas à apresentação visual; ele desempenha um papel crucial na garantia de uma experiência fluida e intuitiva para os visitantes. O foco não se limita apenas à estética, mas também abrange a usabilidade, buscando proporcionar uma interação eficiente.

O desenvolvimento *front-end* utiliza diversas tecnologias e linguagens de programação, como HTML, CSS e JavaScript, para estruturar a informação, estilizar os elementos visuais e proporcionar interatividade, respectivamente. Está em constante evolução para atender às tendências de design e às crescentes demandas por interatividade, responsividade e acessibilidade.

### 2.4.1 HTML

O HTML (*HyperText Markup Language*) emerge como uma linguagem de marcação fundamental e amplamente adotada, estabelecendo-se como o padrão no desenvolvimento de estruturas de conteúdo na web. Ao utilizar marcações específicas, o HTML viabiliza a construção de páginas de websites, permitindo a incorporação de elementos como imagens, vídeos, textos, títulos e subtítulos. Essas marcações conferem uma organização estruturada, facilitando a personalização da página e a integração de outros códigos (Whatwg, 2023).

Além dessa capacidade estrutural, o HTML também possibilita a criação de *links* entre as páginas, propiciando uma transição fluida entre diferentes seções do mesmo site ou até mesmo a navegação para outros websites na internet. Essa habilidade de estabelecer conexões entre as páginas contribui significativamente para a experiência de navegação do usuário, promovendo a acessibilidade e a interconectividade das informações (Mozilla, 2023).

### 2.4.2 CSS

O CSS (*Cascading Style Sheets*) é uma linguagem de estilo essencialmente utilizada para controlar e aprimorar a apresentação de documentos escritos em HTML. Em conjunto com HTML e Javascript, o CSS constitui uma das principais tecnologias na construção de páginas web (Scheidt, 2015).

A função primordial do CSS é viabilizar a personalização de uma ampla variedade de estilos para elementos específicos do HTML, influenciando diretamente a aparência visual de uma página web. Essa personalização abrange diversos aspectos, incluindo a definição de cores, tamanhos de fonte e a escolha de estilos de fonte para o texto em todas as páginas de um projeto. Além disso, o CSS permite configurações específicas, como o design de um título particular, proporcionando um alto grau de flexibilidade no design e estilização de páginas web (Murphy et al., 2012).

### 2.4.3 Javascript

JavaScript (JS) emerge como uma linguagem de programação interpretada leve, notória por sua capacidade de compilação *just-in-time* e caracterizada pela presença de funções de primeira classe. Embora seja predominantemente reconhecida como a linguagem de *script* para

páginas da web, implementada, por exemplo, nos navegadores (Mozilla, 2022), sua versatilidade a tornou uma das linguagens de programação mais populares globalmente.

Conforme apontado por Lin e Gebaly (2016), criada por Brendan Eich em 1995, o JavaScript desempenha um papel fundamental tanto no desenvolvimento *front-end* quanto no *backend*, destacando-se pela integração íntima com HTML e CSS. No contexto do desenvolvimento de páginas web dinâmicas, essa linguagem oferece recursos essenciais que possibilitam a criação de interfaces interativas e responsivas. Sua presença é ubíqua na construção de aplicações modernas na web, onde a interação do usuário é um elemento crucial.

#### **2.4.4 BootStrap**

Conforme destacado no site oficial do Bootstrap (2023), este *framework* de código aberto é extensivamente utilizado para o desenvolvimento de interfaces de usuário web responsivas e móveis. Inicialmente concebido por desenvolvedores do Twitter, o Bootstrap disponibiliza uma coleção de ferramentas baseadas em HTML, CSS e JavaScript, visando simplificar e agilizar o processo de design de sites e aplicações web. O *framework* oferece uma gama abrangente de componentes reutilizáveis, como botões, formulários, navegação, entre outros elementos de interface, previamente estilizados e configurados para assegurar compatibilidade e funcionalidade em diversos dispositivos e tamanhos de tela.

Além de sua ênfase em proporcionar uma experiência de usuário responsiva e móvel, o Bootstrap se destaca pela sua abordagem modular, permitindo aos desenvolvedores utilizar e personalizar componentes de acordo com as necessidades específicas de seus projetos. Esse caráter modular contribui significativamente para a eficiência no desenvolvimento, uma vez que oferece uma estrutura coesa que promove a consistência visual e funcional em todo o projeto.

O Bootstrap também integra um sistema de *grid* responsivo, facilitando a organização e disposição flexível de elementos na página. Esse sistema proporciona uma base sólida para o design, possibilitando a criação de *layouts* fluidos que se adaptam de maneira elegante a diferentes tamanhos de tela.

Dessa forma, o Bootstrap emerge não apenas como uma ferramenta para simplificar a criação de interfaces, mas como um aliado estratégico para desenvolvedores na busca por eficiência, consistência e adaptabilidade em seus projetos web. Seu uso disseminado na comunidade de desenvolvimento é reflexo da sua eficácia e versatilidade na construção de experiências online de alta qualidade.

## 2.5 BACKEND

O *backend*, fundamental para o pleno funcionamento de uma aplicação, desempenha um papel vital ao viabilizar a execução das tarefas apresentadas no *front-end*. Esta camada, responsável pelo processamento efetivo dos dados essenciais, constitui o ambiente onde a aplicação é executada. Dentre os diversos elementos abrangidos pelo *backend*, destacam-se a lógica de programação, servidores, bancos de dados, APIs (Interfaces de Programação de Aplicações) e outros componentes (Filipova et al., 2018).

Essa estrutura é central para a eficiência operacional de uma aplicação, uma vez que engloba a manipulação e processamento de dados, interação com bancos de dados para armazenamento e recuperação de informações, implementação de lógica de negócios, e a disponibilização de APIs para a comunicação tanto com o *front-end* quanto com outros serviços. O *backend*, assim, desempenha um papel indispensável no funcionamento harmônico da aplicação, assegurando o processamento eficaz das ações solicitadas pelo usuário no *front-end* e a apresentação dos resultados de forma coerente ao usuário.

## 2.6 SELENIUM

Conforme o site oficial do Selenium (2023), este projeto compreende uma variedade de ferramentas e bibliotecas dedicadas à automação de navegadores web. O Selenium oferece extensões que possibilitam a emulação da interação do usuário com os navegadores, além de um servidor de distribuição que permite a escalabilidade na alocação dos navegadores. Adicionalmente, o Selenium oferece suporte à implementação da especificação W3C *WebDriver*, proporcionando a capacidade de escrever código intercambiável para todos os principais navegadores da web.

No cerne do Selenium encontra-se o *WebDriver*, uma interface que viabiliza a redação de conjuntos de instruções executáveis de maneira interoperável em diversos navegadores. O site também disponibiliza uma visão abrangente dos diferentes componentes do projeto, bem como um guia introdutório para facilitar a instalação e utilização do Selenium como ferramenta de automação de testes. Essa ferramenta é capaz de dimensionar testes desde os mais simples

até ambientes robustos e distribuídos, abrangendo diferentes navegadores e sistemas operacionais.

## 2.7 BANCO DE DADOS

Conforme o website da Oracle (2023), um banco de dados é uma coleção organizada de informações ou dados, geralmente armazenados eletronicamente em um sistema de computador. Essas informações são estruturadas e organizadas de maneira a facilitar o processamento eficiente dos dados. A organização padrão de um banco de dados ocorre em tabelas, com dados dispostos em colunas e linhas, proporcionando fácil acesso, manutenção, modificação, atualização, controle e organização dos dados.

De acordo com Elmasri e Navathe (2011), existem várias abordagens para organizar um banco de dados, sendo algumas das principais:

- Relacional: Os dados são organizados em tabelas e colunas. Este modelo é conhecido como banco de dados relacional e utiliza relacionamentos entre tabelas para representar a estrutura dos dados;
- Relacional a Objeto: Neste modelo, os dados são representados na forma de objetos, refletindo estruturas mais complexas e permitindo uma modelagem mais próxima do mundo real;
- Distribuído: Permite a distribuição de dados em arquivos que podem estar localizados em diferentes locais. Esses arquivos são, então, integrados na mesma base de dados, facilitando a gestão de dados em ambientes distribuídos;
- NoSQL ou Não Relacional: Este modelo permite a manipulação de dados não organizados ou sem uma estrutura fixa. É particularmente adequado para ambientes nos quais a flexibilidade na gestão de dados é fundamental.

Essas diferentes formas de organização de banco de dados atendem a necessidades específicas de aplicação e casos de uso. A escolha entre elas depende dos requisitos do projeto, da natureza dos dados e das metas de desempenho e escalabilidade.

### 2.7.1 MongoDB

O MongoDB é um banco de dados projetado para proporcionar escalabilidade e flexibilidade, oferecendo ferramentas fundamentais para aplicações modernas. Ao contrário dos bancos de dados relacionais tradicionais, o MongoDB é categorizado como um banco de dados NoSQL, indicando que não adere à estrutura convencional de tabelas e colunas característica dos bancos relacionais (Awari, 2022).

Conforme detalhado pelos autores Hows, Membrey e Plugge, no MongoDB, os dados são armazenados em documentos flexíveis no formato JSON (*JavaScript Object Notation*). Essa abordagem possibilita que os desenvolvedores trabalhem de maneira mais intuitiva e eficiente com os dados relacionados aos seus projetos. No MongoDB, cada documento é uma representação em JSON que contém pares chave-valor, oferecendo uma estrutura semelhante à de objetos JavaScript.

A flexibilidade no armazenamento de dados e o uso de documentos JSON tornam o MongoDB especialmente adequado para cenários nos quais a estrutura dos dados pode variar ou crescer ao longo do tempo. Adicionalmente, o MongoDB é reconhecido por sua capacidade de escalar horizontalmente, o que significa que pode lidar com grandes volumes de dados distribuídos em vários servidores, mantendo a flexibilidade e o desempenho. Essas características fazem do MongoDB uma escolha relevante para aplicações que demandam agilidade, evolução constante e gerenciamento eficaz de grandes conjuntos de dados.

### 2.7.2 MongoDB Atlas

De acordo com no website do MongoDB Atlas (2023), o MongoDB Atlas é um serviço de banco de dados em nuvem desenvolvido pela mesma equipe responsável pelo MongoDB. Esse serviço simplifica tanto a implementação quanto a administração de bancos de dados, proporcionando a versatilidade necessária para criar aplicativos globais resilientes e de alto desempenho nos provedores de nuvem escolhidos.

O MongoDB Atlas opera como um *Database-as-a-Service* (DBaaS), sistema que permite configurar, implantar e dimensionar um banco de dados sem a necessidade de lidar com hardware físico local, atualizações de software e detalhes de configuração para otimização de desempenho. Com o DBaaS, um provedor de nuvem realiza essas tarefas de maneira automática, oferecendo uma solução mais eficiente e simplificada para as necessidades de gerenciamento de banco de dados em ambientes de nuvem.

## 2.8 NODE.JS E NODE PACKAGE MANAGER (NPM)

Conforme Tilkov e Vinoski (2010), o Node.js, ou simplesmente Node, é um sistema de código aberto amplamente adotado por desenvolvedores para executar códigos *JavaScript* fora do navegador do cliente, ou seja, no lado do servidor. Introduzido por Ryan Dahl em 2009, o Node.js destaca-se como uma escolha popular para o desenvolvimento de aplicativos em tempo real. Sua notoriedade é amplificada pela presença de uma extensa comunidade de desenvolvedores que contribuem ativamente para o seu aprimoramento contínuo.

Uma característica proeminente do ecossistema Node.js é o NPM (*Node Package Manager*), um gerenciador de pacotes que facilita a instalação e distribuição de códigos específicos para o Node.js. Essa ferramenta desempenha um papel crucial no desenvolvimento ao permitir o eficiente gerenciamento de dependências e bibliotecas em uma variedade de projetos. Através do NPM, os desenvolvedores podem integrar facilmente soluções de terceiros, simplificando o processo de desenvolvimento e aprimorando a eficiência na gestão de projetos Node.js (SOUZA, 2020).

## 2.9 VISUAL STUDIO CODE

De acordo com o website do Visual Studio Code (2023), este é uma ferramenta de edição de código gratuita e de código aberto desenvolvida pela Microsoft®. Ele se destaca por oferecer suporte a diversas linguagens de programação, abrangendo desde C, C# e Java até Javascript, Python e várias outras. Sua versatilidade e flexibilidade fazem dele uma escolha popular entre desenvolvedores de diferentes plataformas.

Além disso, o Visual Studio Code possui uma ampla gama de extensões e *plugins* que podem ser facilmente integrados, permitindo aos desenvolvedores personalizar a experiência de codificação de acordo com suas necessidades específicas. Sua interface intuitiva e recursos avançados, como depuração integrada, controle de versionamento e suporte a Git, contribuem para tornar o processo de desenvolvimento mais eficiente e produtivo. O Visual Studio Code é uma ferramenta abrangente que atende tanto iniciantes quanto profissionais experientes, consolidando sua posição como uma das principais escolhas no cenário de desenvolvimento de *software*.

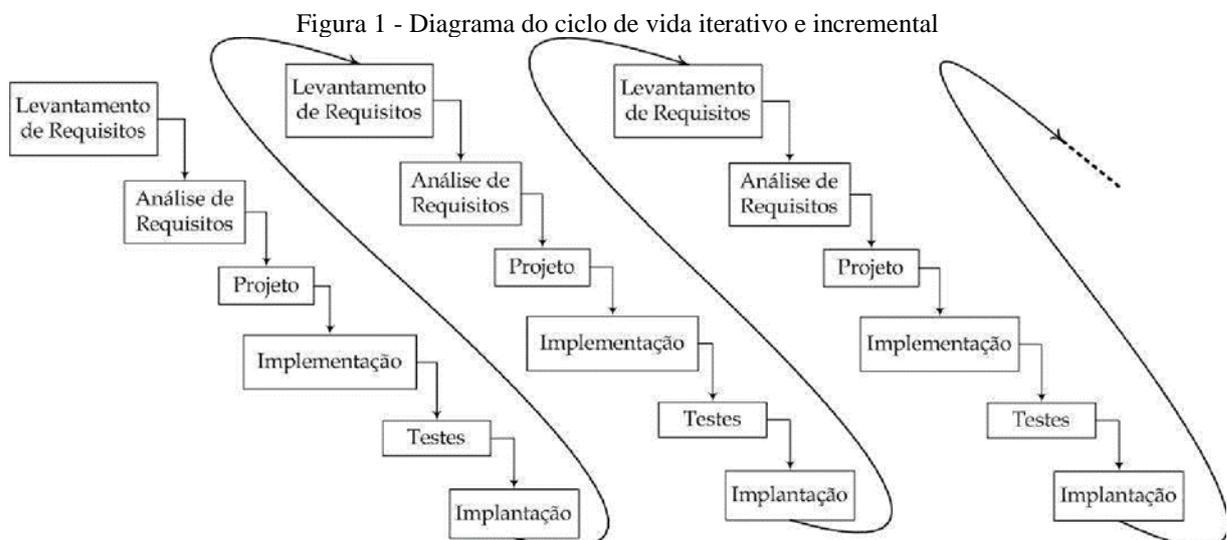
### 3 METODOLOGIA

Este capítulo aborda os métodos que foram utilizados para o desenvolvimento da aplicação prática.

#### 3.1 MODELO ITERATIVO E INCREMENTAL

O modelo iterativo e incremental representa uma metodologia de desenvolvimento de software que integra características dos modelos iterativo e cascata, proporcionando flexibilidade e adaptabilidade no decorrer de projetos de software. Nesse modelo, o desenvolvimento é segmentado em pequenos incrementos, nos quais cada incremento adiciona funcionalidades ao *software* em desenvolvimento. Essa abordagem possibilita ajustes e melhorias contínuas, sendo especialmente adequada para projetos dinâmicos e sujeitos a mudanças constantes, como é o caso em sistemas de gerenciamento de marketplace (Sommerville, 2011).

O diagrama apresentado na Figura 1 apresenta o processo adotado para a criação da aplicação, que se desenrola da seguinte maneira:



Fonte: IFSC( 2006).

O processo de desenvolvimento da aplicação adota as seguintes etapas:

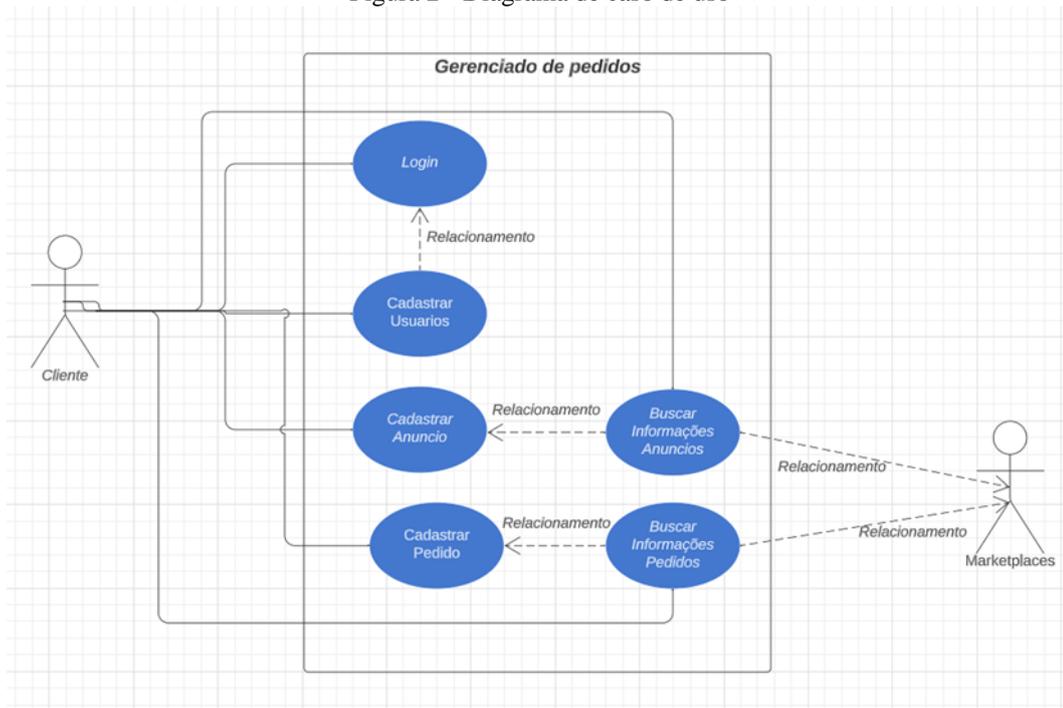
- Coleta de Requisitos: Nesta fase, busca-se identificar as necessidades e problemas dos usuários, examinando as práticas atuais e identificando oportunidades de melhoria. O objetivo é definir as especificações do software a ser criado;
- Análise de Requisitos: Nesta etapa, ocorre a interpretação e avaliação dos requisitos coletados, visando propor soluções viáveis para atender às demandas identificadas;
- Projeto: Na fase de projeto, são escolhidas as ferramentas necessárias para o desenvolvimento e ocorre a estruturação da aplicação, delineando como ela será implementada;
- Implementação: A etapa de implementação é dedicada ao desenvolvimento efetivo das funcionalidades da aplicação, colocando em prática o que foi definido nas fases anteriores;
- Testes: Consiste na verificação das funcionalidades desenvolvidas para identificar e corrigir eventuais falhas. Esses testes podem ser realizados em ambientes controlados, com ou sem a participação do usuário final;
- Lançamento: Finalmente, na última etapa, as funcionalidades testadas e aprovadas são implementadas no ambiente dos usuários finais, tornando a aplicação disponível para uso.

### 3.2 LEVANTAMENTO E ANÁLISE DE REQUISITOS

Os requisitos constituem um conjunto de diretrizes estabelecidas pelos usuários, delineando a estrutura e o comportamento desejado para o software em desenvolvimento. Isso engloba aspectos como o processo, os dados gerados esperados, as restrições de funcionamento e os atores envolvidos, abordando de forma abrangente todos os problemas relacionados (Canguçu, 2021). O processo de levantamento dos requisitos foi realizado em colaboração com os usuários do *marketplace*, visando compreender suas necessidades específicas para o projeto.

A Figura 2 apresenta o diagrama de casos de uso planejado para a utilização do sistema, proporcionando uma visualização clara das interações previstas entre os usuários e o sistema em questão.

Figura 2 - Diagrama de caso de uso



Fonte: Autor.

### 3.2.1 Requisitos funcionais

Os requisitos funcionais do sistema são apresentados no Quadro 1.

Quadro 1 - Requisitos Funcionais

RF[01]	Gerenciar pedidos e estoques
RF[02]	Automatizar o processo de recebimento, confirmação de pedidos
RF[03]	Integrar com várias plataformas de <i>e-commerce</i> e <i>marketplaces</i>
RF[04]	Gerar relatórios e análises de vendas

Fonte: Autor.

### 3.2.2 Requisitos não funcionais

Os requisitos não funcionais do sistema são apresentados no Quadro 2.

Quadro 2 - Requisitos Não Funcionais

RNF[01]	Compatibilidade com diferentes sistemas operacionais
RNF[02]	O sistema deverá informar onde cada anúncio está ativo
RNF[03]	Coleta de informações automática dos marketplace
RNF[04]	A aplicação não deverá ter custos de operação.
RNF[05]	Implementação de medidas de segurança para proteger dados dos usuários
RNF[06]	Escalabilidade do sistema para suportar o crescimento do volume de pedidos e usuários
RNF[07]	Interface responsiva, garantindo usabilidade em dispositivos móveis e desktops

Fonte: Autor.

## 4 RESULTADOS E DISCUSSÕES

Neste capítulo serão apresentados os principais resultados obtidos com o desenvolvimento do protótipo do sistema.

### 4.1 LEVANTAMENTO DE REQUISITOS

A fase inicial do projeto envolveu um levantamento de requisitos realizado por meio de entrevistas com um grupo focal pequeno, composto por dois vendedores e dois consumidores ativos no ambiente de marketplaces. Essas conversas, apesar de serem informais, foram fundamentais para obter uma compreensão inicial das necessidades e expectativas dos usuários. Durante esses encontros, foi possível identificar aspectos-chave relacionados à usabilidade, funcionalidade e experiência geral de usuário desejada para o sistema.

Apesar da amostra limitada, as informações coletadas forneceram *insights* valiosos para a concepção do projeto. As discussões focaram em identificar problemas comuns enfrentados pelos usuários na gestão de pedidos e na experiência de compra online. As percepções obtidas dessas conversas foram cuidadosamente documentadas e analisadas, formando a base para o desenvolvimento das especificações iniciais do sistema. Esta abordagem direcionada e focada em usuários reais assegurou que o projeto começasse com uma sólida compreensão das necessidades do mercado, preparando o terreno para as fases subsequentes de desenvolvimento.

### 4.2 VISÃO GERAL DO SISTEMA

No processo de desenvolvimento do sistema, optou-se por utilizar as principais ferramentas, sendo o *MongoDB Atlas* e o *Visual Studio Code*. O *MongoDB Atlas* desempenhou um papel crucial no armazenamento e gerenciamento eficiente dos dados, proporcionando um registro seguro das informações utilizadas no sistema.

Quanto à edição de códigos, o *Visual Studio Code* foi a escolha por oferecer um ambiente de desenvolvimento rico em recursos, com suporte a depuração, controle de versão

integrado e uma vasta biblioteca de extensões. Sua flexibilidade e capacidade de suporte a várias linguagens de programação o tornam ideal para o desenvolvimento de um sistema complexo e multifacetado. O Visual Studio Code também se destaca por sua interface amigável e personalizável, permitindo que os desenvolvedores configurem o ambiente de acordo com suas preferências e necessidades.

No âmbito do *backend*, o sistema foi desenvolvido utilizando Node.js, que foi escolhido por ser uma plataforma *backend* eficiente e leve. Sua arquitetura orientada a eventos é particularmente adequada para aplicações que exigem um alto volume de operações de entrada/saída, como é o caso de sistemas que interagem com múltiplos marketplaces. Além disso, o Node.js se beneficia de um vasto ecossistema de módulos disponíveis via *npm*, aumentando ainda mais a produtividade e a flexibilidade no desenvolvimento. A linguagem *JavaScript* foi escolhida pela sua ampla aplicabilidade tanto no *backend* quanto no *frontend* do sistema.

No desenvolvimento do *frontend*, além dos scripts em *JavaScript*, foram empregados HTML e o *Bootstrap*. Essas tecnologias foram selecionadas visando criar uma interface de usuário refinada e agradável, simplificando consideravelmente o processo de desenvolvimento. O *Bootstrap*, com seus componentes reutilizáveis e design responsivo, contribuiu para uma experiência de usuário mais intuitiva e acessível em diferentes dispositivos.

Para a coleta de dados dos *marketplaces* disponíveis no mercado, a ferramenta *Selenium* foi utilizada para a coleta de dados devido à sua capacidade de automatizar navegadores web de forma eficiente. O *Selenium* permite simular interações do usuário em diferentes plataformas de *e-commerce* de através de *scripts* configuráveis, facilitando a coleta e integração de dados. Esta ferramenta é indispensável para garantir que o sistema mantenha atualizações constantes e precisas dos dados dos marketplaces, essencial para a eficiência operacional do sistema.

### 4.3 ANÁLISE DAS PÁGINAS FUNCIONAIS

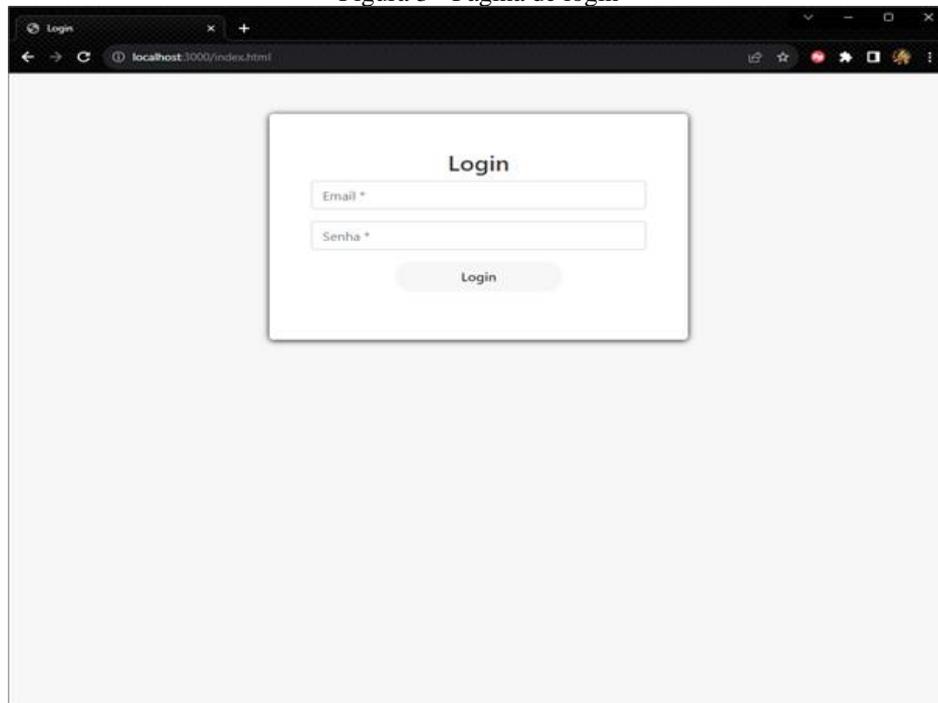
Nas próximas seções serão apresentadas as páginas do protótipo desenvolvido.

#### 4.3.1 Login

Descrição: Inclui na página de login, campo para e-mail e senha.

**Importância:** A página de login é fundamental para a segurança e acessibilidade do sistema, garantindo que apenas usuários autorizados possam acessar o sistema fazendo uma conferência com o banco de dados para verificar se os dados estão cadastrados. Na Figura 3 é apresentada a primeira página que usuário vê quando acessa o sistema.

Figura 3 - Página de login



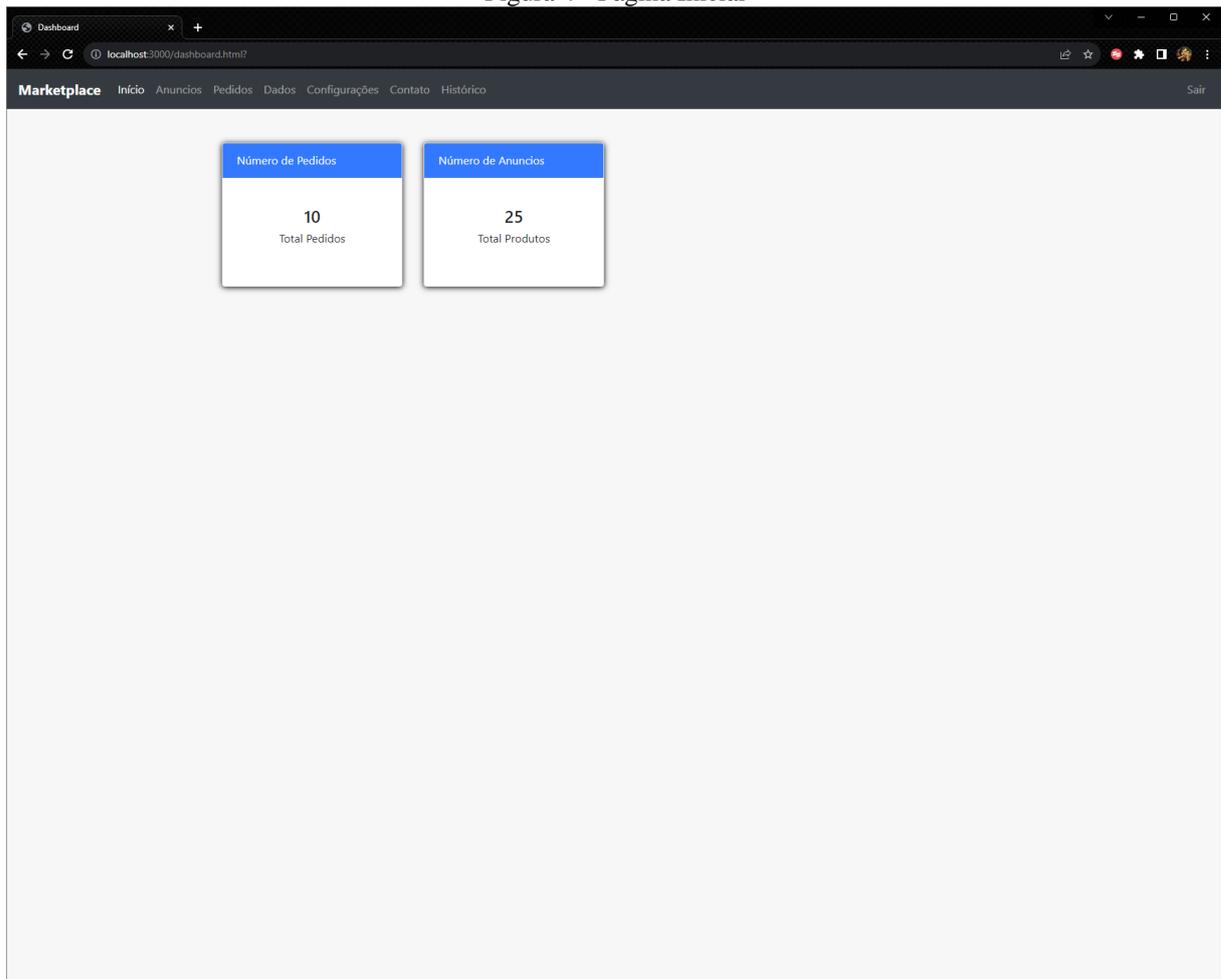
Fonte: Autor.

### 4.3.2 Home

**Descrição:** Apresentação de Informações Básicas. Os usuários podem visualizar um resumo rápido dos pedidos pendentes e a quantidade de itens.

**Importância:** a página oferece uma visão geral imediata do estado atual dos negócios, permitindo aos usuários tomarem decisões informadas rapidamente e identifiquem áreas que necessitam de atenção. A Figura 4 apresenta a página desenvolvida.

Figura 4 - Página Inicial



Fonte: Autor.

### 4.3.3 Gestão de Anúncios

**Descrição:** Esta página é dedicada ao gerenciamento de produtos. Aqui, os usuários podem adicionar novos produtos, verificar o preço e apagar produtos antigos. Esses itens são cadastrados diretamente pelo usuário para controle, ou podem ser adicionados no botão de *update*.

**Importância:** Manter informações de produto atualizadas e precisas é fundamental. Esta página auxilia na gestão eficiente do inventário, evitando problemas e discrepâncias de informações.

Figura 5 - Página Gestão de Anúncios

Gerenciador de Anúncios

**Adicionar produto**

Nome

Preço

Quantidade

Buscar no marketplace

**Buscar no marketplace**

Selecionar Marketplace:

Lista de produtos

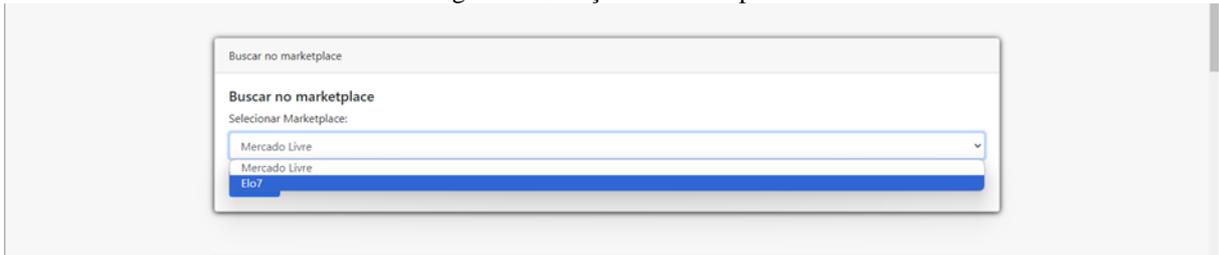
**Lista de produtos**

Kit C/3 Fio Duplo Em Poliam E Renda Tam. P - R\$ 49,90 - Elo7 - Quantidade:	<input type="text" value="1"/>	<input type="button" value="Excluir"/>
CJ Sainha C/ Liga Em Cirre E Rendaref: 5175 Tam. M - R\$ 150,00 - Elo7 - Quantidade:	<input type="text" value="1"/>	<input type="button" value="Excluir"/>
Pijama manga longa flamingo tam. G - R\$ 95,00 - Elo7 - Quantidade:	<input type="text" value="1"/>	<input type="button" value="Excluir"/>
Pijama manga longa flamingo tam. M - R\$ 95,00 - Elo7 - Quantidade:	<input type="text" value="1"/>	<input type="button" value="Excluir"/>

Fonte: Autor.

Descrição: A opção de “Buscar no marketplace” ativa o *script* do *Selenium* para fazer a coleta dos itens diretamente no site da empresa dos diferentes *marketplaces*. Para adicionar de forma automatizada, ao clicar no botão *update*, a aplicação busca nos diferentes *marketplaces* as informações dos anúncios. Para este trabalho, integrou-se com dois marketplaces, sendo: Mercado Livre® e Elo7®. Para que essa atualização ocorra e por medidas de segurança, ao selecionar o *marketplace* que deseja coletar as informações dos anúncios, o usuário insere as credenciais cadastradas no *marketplace* específico. Após a validação, a tela é atualizada com as informações coletadas, conforme apresentado na Figura 6.

Figura 6 - Seleção de marketplace



Fonte: autor.

O sistema utiliza o *Selenium WebDriver* para automatizar a coleta de dados dos marketplaces. Quando um marketplace específico é selecionado, o servidor ativa um *script* personalizado do *Selenium* que acessa automaticamente a página da loja online. Uma vez que o usuário insere suas credenciais de login, o *Selenium* executa a tarefa de extrair informações relevantes da página, conforme definido previamente no script. Este processo inclui a captura de detalhes críticos dos anúncios, como preços, descrições de produtos. As informações coletadas são então enviadas para o banco de dados do sistema. Essa abordagem automatizada não só economiza tempo, mas também reduz o risco de erros manuais, garantindo que os dados coletados sejam precisos e atualizados.

A lista de produtos irá informar todos os itens que o usuário tem anunciado, e na própria página tem a opção de alterar a quantidade para manter o controle de estoque ou excluir o item anunciado, por exemplo, quando não estiver mais cadastrado nos sistemas. Também irá mostrar onde cada anúncio está publicado. A Figura 7 apresenta a lista de produtos.

Figura 7 - Lista de produtos

Cesto de palha natural taboa 35 cm oval - R\$ 32,40 - Elo7 - Quantidade: <input type="text" value="1"/>	Excluir
Kit 2 Cesto de palha taboa 34 cm - R\$ 95,00 - Elo7 - Quantidade: <input type="text" value="1"/>	Excluir
Cesto de palha 30 cm - R\$ 32,40 - Elo7 - Quantidade: <input type="text" value="1"/>	Excluir
Bandeja de palha natural taboa - R\$ 97,20 - Elo7 - Quantidade: <input type="text" value="1"/>	Excluir
Pantufa Com Borda, Referencia: 3392 - R\$ 102,50 - Elo7 - Quantidade: <input type="text" value="1"/>	Excluir
Kit cachepo de macrame com vasinho 5 cm - R\$ 16,20 - <u>Elo7</u> - Quantidade: <input type="text" value="1"/>	Excluir
Kit Cachepo em macrame mais vasinho em cerâmica - R\$ 21,60 - <u>Elo7</u> - Quantidade: <input type="text" value="1"/>	Excluir
Notebook Asus Zenbook 13 Oled Ux325 I5 8gb 256gb W11 Cinza - R\$ 3.000 - <u>Mercado Livre</u> - Quantidade: <input type="text" value="1"/>	Excluir
Smartphone Samsung Galaxy S20 Tela 6.2 128gb 8gb Ram - R\$ 2.500 - <u>Mercado Livre</u> - Quantidade: <input type="text" value="1"/>	Excluir

Fonte: Autor.

### 4.3.4 Pedidos

Descrição: Esta página permite aos usuários visualizarem, processarem e gerenciarem todos os pedidos recebidos dos *marketplaces*. Assim como nos produtos os pedidos podem ser colocados de forma manual ou de forma automatizada utilizando o botão de *update*.

Importância: A eficiência na gestão de pedidos é crucial para o sucesso de negócios de *e-commerce*. Esta página torna o processo de gerenciamento de pedidos mais organizado e menos propenso a erros, melhorando a satisfação do cliente.

Nessa janela o cadastro será de pedidos que o usuário ainda tem em aberto, e que tem que fazer a entrega, ou esperar a confirmação de entrega da empresa. Da mesma forma como ocorre na gestão de anúncios, o anúncio pode ser preenchido de forma manual para controle do usuário, ou pode ser inserida de forma automática buscando nos diferentes *marketplaces*. A Figura 8 apresenta a página de pedidos.

Figura 8 - Página de Pedidos

The screenshot shows a web browser window with the URL 'localhost:3000/orders.html'. The page title is 'Marketplace' and the navigation menu includes 'Início', 'Anúncios', 'Pedidos', 'Dados', 'Configurações', 'Contato', and 'Histórico'. The main content area is divided into three sections:

- Adicionar Pedidos:** A form with fields for 'Pedido', 'Quantidade', and 'Status' (set to 'Pendente'), and a blue 'Adicionar' button.
- Buscar no marketplace:** A section titled 'Buscar no marketplace' with a dropdown menu set to 'Mercado Livre' and a blue 'Buscar' button.
- Lista de Pedidos:** A table listing six orders with their IDs, quantities, store names, and statuses. Each row has a dropdown menu for the status and a grey 'Excluir' button.
 

Pedido	Quantidade	Loja	Status	Ação
Pedido: Pedido 2F8F065	1 produto	Loja: Elo7	Entregue	Excluir
Pedido: Pedido 2F4846F	1 produto	Loja: Elo7	Entregue	Excluir
Pedido: Pedido 2E20C74	1 produto	Loja: Elo7	Entregue	Excluir
Pedido: Pedido 2E187C3	1 produto	Loja: Elo7	Enviado	Excluir
Pedido: Pedido 2C02898	1 produto	Loja: Elo7	Enviado	Excluir
Pedido: Pedido 2A7A15F	1 produto	Loja: Elo7	Enviado	Excluir

Fonte: Autor.

Na lista de produtos será gravado o atual status coletado no *marketplace*, após isso o cliente tem a possibilidade de alterar o status diretamente na lista para o controle próprio e/ ou excluir o registro do banco de dados diretamente na lista.

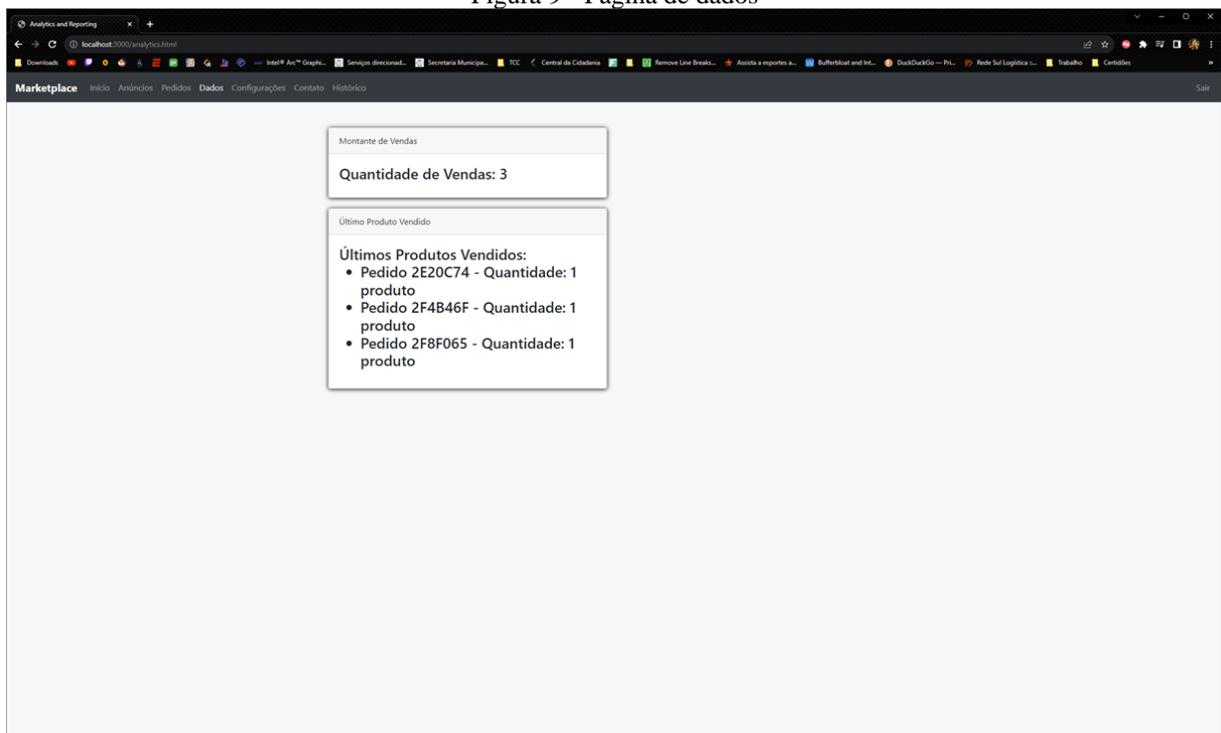
O sistema também emprega um script de *Selenium WebDriver* para a automação do processo de gerenciamento de pedidos. De maneira semelhante à coleta de dados de anúncios, o *Selenium* é configurado para acessar os marketplaces e coletar informações detalhadas sobre os pedidos em aberto, incluindo status de entrega. Esta funcionalidade permite que os usuários do sistema tenham uma visão atualizada e automatizada dos pedidos, refletindo as alterações de status nos marketplaces. A integração desse script de Selenium aprimora significativamente a eficiência operacional do sistema, reduzindo a necessidade de entrada manual de dados e proporcionando um controle mais preciso e atualizado dos pedidos.

### 4.3.5 Dados

**Descrição:** Esta página fornecerá análises detalhadas e relatórios de vendas, tendências de mercado, e comportamento do cliente, os dados serão analisados conforme os clientes vão colocando as informações.

**Importância:** A análise de dados e relatórios detalhados são essenciais para estratégias de negócios e tomadas de decisão baseadas em evidências. A Figura 9 apresenta a página criada.

Figura 9 - Página de dados



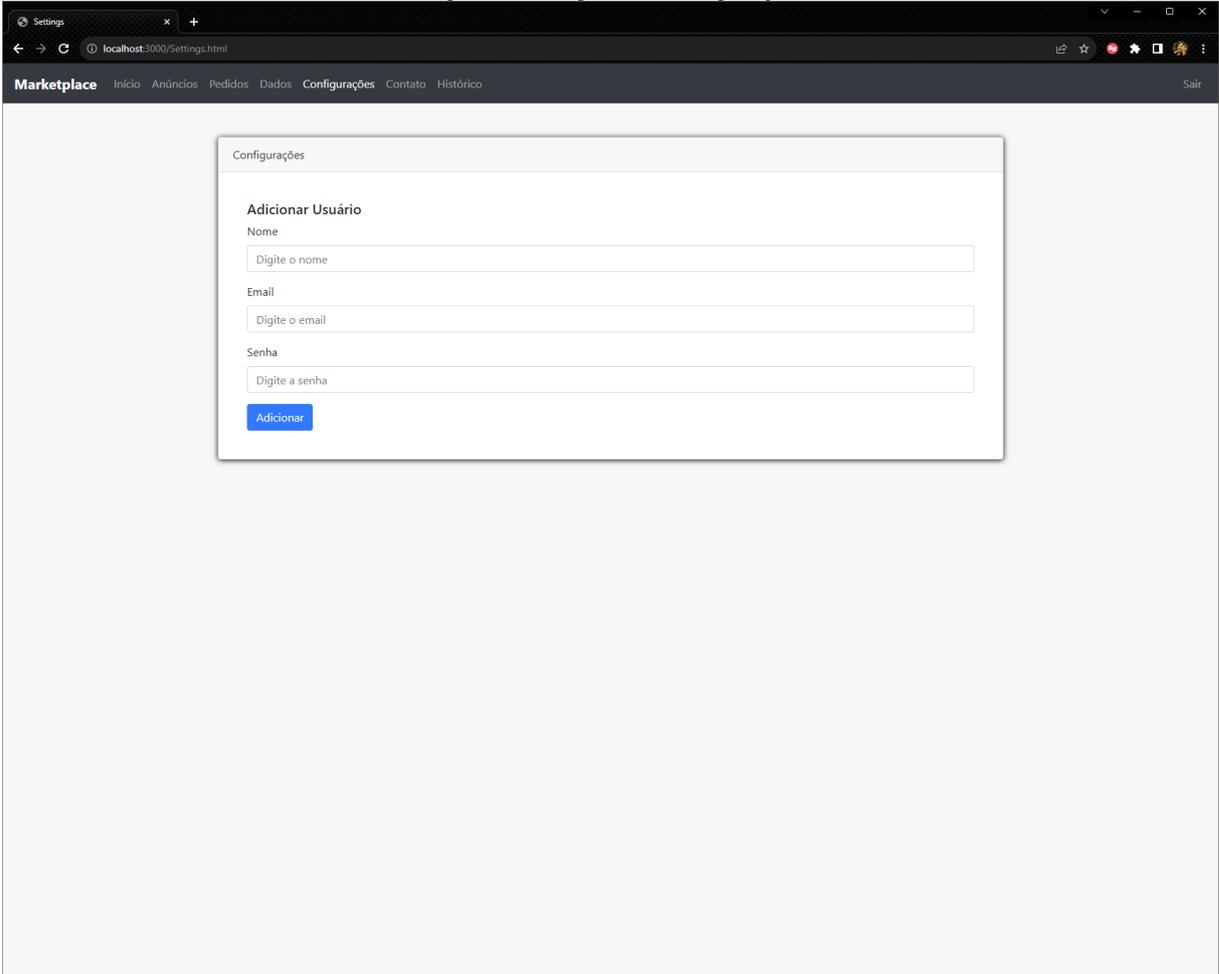
Fonte: Autor.

### 4.3.6 Configurações

Descrição: Esta página permitirá aos usuários cadastrar novos usuários.

Importância: Permite ao usuário adicionar novos utilizadores para o sistema. A Figura 10 apresenta a página desenvolvida para possibilitar configurações.

Figura 10 - Página de configuração



The screenshot shows a web browser window with the address bar displaying 'localhost:3000/Settings.html'. The browser's navigation bar includes the text 'Marketplace' and a menu with items: 'Início', 'Anúncios', 'Pedidos', 'Dados', 'Configurações', 'Contato', and 'Histórico'. A 'Sair' link is visible on the right. The main content area features a form titled 'Configurações' with the sub-heading 'Adicionar Usuário'. The form contains three input fields: 'Nome' with the placeholder 'Digite o nome', 'Email' with 'Digite o email', and 'Senha' with 'Digite a senha'. A blue 'Adicionar' button is positioned below the 'Senha' field.

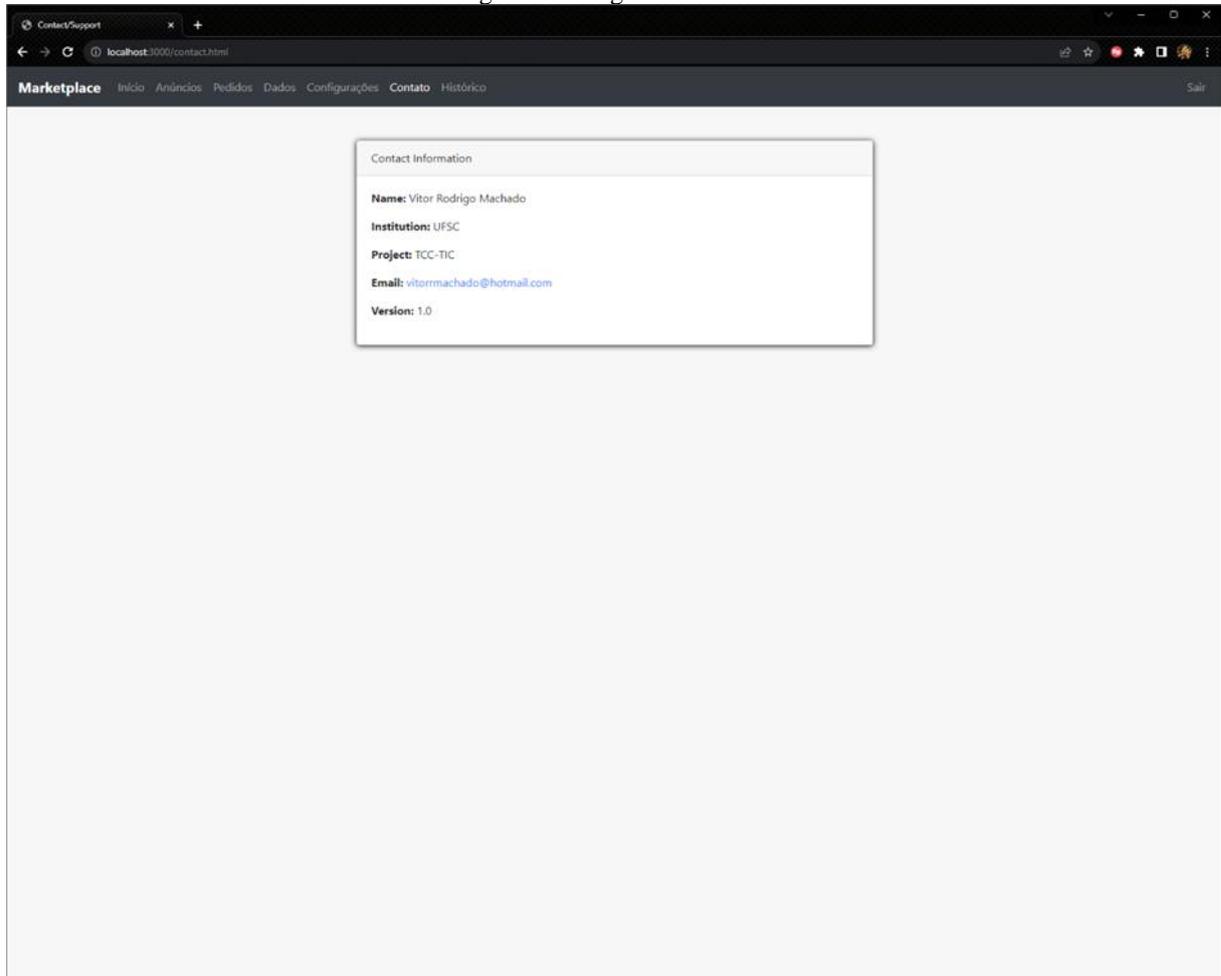
Fonte: Autor.

### 4.3.7 Contato

Descrição: Esta página oferece informações do sistema e da equipe de desenvolvimento, além de fornecer contato para caso precise de suporte.

Importância: Um suporte ao cliente eficaz é crucial para manter a confiança e satisfação do usuário, permitindo a resolução rápida de problemas e dúvidas. A Figura 11 apresenta a página de contato.

Figura 11 - Página de contato



Fonte: Autor.

#### 4.4 FUNCIONALIDADES EM DESENVOLVIMENTO

À medida que o sistema de gerenciamento de pedidos continua a evoluir, mais funcionalidades podem ser implementadas ao sistema. As seguintes funcionalidades futuras estão previstas para integrar o projeto:

- Integração com Plataformas de Mídias Sociais e *Marketing*;
- Automação de Marketing: A integração com ferramentas de marketing digital permitirá a automação de campanhas publicitárias, postagens em mídias sociais e

envio de *emails* promocionais, diretamente vinculadas aos dados de produtos e vendas do sistema;

- Funcionalidades de Personalização para Usuários;
- *Dashboard* Personalizável: Permitir que os usuários personalizem seus *dashboards* para destacar as informações mais relevantes para suas necessidades específicas;
- Relatórios Personalizados: Os usuários poderão gerar relatórios personalizados baseados em critérios específicos, ajudando na análise detalhada de aspectos distintos do negócio;
- Segurança e Conformidade Aprimoradas;
- Adaptação para Novos Marketplaces: Expandir a capacidade do sistema para integrar-se com uma variedade ainda maior de *marketplaces*, incluindo aqueles em mercados internacionais emergentes;
- Suporte multilíngue: Implementar suporte multilíngue para facilitar o uso do sistema em diferentes regiões do mundo;
- Análises Avançadas e Relatórios de Tendências;
- Análise Preditiva: Utilizar dados históricos para prever tendências futuras de vendas e comportamento do consumidor.

#### 4.5 ARMAZENAMENTO E APRESENTAÇÃO DE DADOS

O *MongoDB Atlas* é uma parte integral do sistema, atuando como a espinha dorsal para o armazenamento e gerenciamento de dados. Possuindo as seguintes características:

- Flexibilidade de Esquema: Uma das principais vantagens do *MongoDB Atlas* é sua natureza *NoSQL*, que permite um esquema de dados flexível. Isso é particularmente útil em um ambiente de *e-commerce* devido à natureza diversificada e dinâmica dos dados envolvidos. Em primeiro lugar, o *e-commerce* lida com uma ampla gama de informações, incluindo detalhes de produtos, estoque, status de pedido. Cada uma dessas categorias pode ter seus próprios atributos únicos e variáveis, o que torna os esquemas rígidos e pré-definidos inadequados.

- Com o MongoDB Atlas, não há necessidade de definir previamente a estrutura do banco de dados. Isso permite que novos tipos de dados sejam incorporados facilmente, sem a necessidade de reestruturar ou modificar as tabelas existentes. Essa flexibilidade é necessária para um setor que está constantemente evoluindo, com novos produtos e categorias sendo adicionados regularmente.
- Escalabilidade e Desempenho: O *MongoDB Atlas* oferece uma escalabilidade automática, garantindo que o sistema possa lidar com o aumento do volume de dados sem comprometer o desempenho. Isso é crucial para garantir uma experiência de usuário consistente, mesmo durante picos de tráfego ou expansão do negócio;
- Segurança de Dados: A segurança é uma prioridade, e o *MongoDB Atlas* oferece recursos robustos, incluindo criptografia em repouso e em trânsito, controles de acesso baseados em função e monitoramento constante, para proteger contra acessos não autorizados e ataques cibernéticos.

#### 4.6 CONSIDERAÇÕES FINAIS

Neste capítulo, foram exploradas as várias etapas e componentes críticos envolvidos no desenvolvimento do protótipo do sistema de gerenciamento de pedidos. Desde o levantamento inicial de requisitos, que proporcionou *insights* valiosos através de entrevistas com vendedores e consumidores, até a escolha cuidadosa de ferramentas como *MongoDB Atlas*, *Visual Studio Code*, *Node.js*, e *Selenium*, cada etapa contribuiu significativamente para a construção de um sistema robusto e eficiente.

As páginas funcionais do sistema, incluindo login, home, gestão de anúncios e pedidos, demonstraram a eficácia do sistema em oferecer uma gestão de *e-commerce* simplificada e automatizada. A implementação do *Selenium WebDriver* para automatização da coleta de dados de marketplaces enfatizou a eficiência e a precisão do sistema. Além disso, o uso do *MongoDB Atlas* ressaltou a importância da flexibilidade de esquema e da segurança de dados em aplicações de *e-commerce*.

Concluindo, este capítulo destaca não só o sucesso técnico do protótipo desenvolvido, mas também estabelece uma base sólida para futuras melhorias e expansões. A integração prevista de funcionalidades adicionais, como automação de marketing e análises avançadas,

aponta para o potencial do sistema em se adaptar e evoluir com as demandas crescentes do mercado de *e-commerce*. Este protótipo é um avanço na criação de sistemas de gerenciamento de pedidos que são mais práticos e automatizados, buscando atender melhor às expectativas atuais do mercado.

## 5 CONSIDERAÇÕES FINAIS

O desenvolvimento do protótipo do sistema de gerenciamento de pedidos para *marketplaces* alcançou resultados significativos, atendendo aos objetivos propostos. O levantamento de requisitos, conduzido em colaboração com usuários e vendedores, estabeleceu parâmetros iniciais fundamentais para a concepção da aplicação. A análise de requisitos, inserida na primeira iteração do modelo iterativo e incremental, guiou o formato e as funcionalidades do sistema.

A escolha estratégica de tecnologias e ferramentas, como *MongoDB Atlas*, *Visual Studio Code* e *Node.js*, destacou-se na implementação das funcionalidades essenciais. O *MongoDB Atlas* desempenhou um papel crucial no armazenamento e gerenciamento eficiente de dados, oferecendo flexibilidade de esquema, escalabilidade e robustos recursos de segurança.

A visão geral do sistema, abrangendo funcionalidades como login, gestão de anúncios, pedidos, dados, configurações e contato, reflete uma abordagem centrada na experiência do usuário. A integração do Selenium para coleta de dados de diferentes *marketplaces* contribuiu significativamente para a atualização automática e precisa das informações, ampliando a funcionalidade do sistema.

As páginas funcionais apresentaram uma interface intuitiva e eficiente. A análise detalhada dessas páginas evidenciou a importância de cada uma na eficiência operacional e na melhoria da satisfação do cliente.

Durante o desenvolvimento do sistema, vários obstáculos foram enfrentados. Um dos principais desafios foi a integração com diversas plataformas de *e-commerce*, exigindo a análise individual entre as escolhidas para conseguir fazer as conexões sem erros.

Esse projeto proporcionou uma solução unificada para o gerenciamento de pedidos de múltiplas plataformas, ele simplifica e otimiza os processos para os comerciantes, uma vez que a automatização de várias tarefas reduz o tempo e o esforço necessários para gerenciar operações de *e-commerce*.

O sistema demonstrou ser capaz de coletar informações de pedidos e anúncios de múltiplos *marketplaces* de forma eficaz, garantindo a precisão dos dados e uma experiência de usuário contínua e sem interrupções. As tecnologias escolhidas para o desenvolvimento, como o *MongoDB Atlas* e *Node.js*, provaram ser adequadas para as necessidades do projeto, oferecendo escalabilidade e desempenho.

Olhando para o futuro, sugestões para aprimoramento incluem a integração com plataformas de mídias sociais e marketing, a automação de marketing, funcionalidades de personalização para usuários, suporte multilíngue e análises preditivas. Essas adições planejadas visam ampliar as capacidades do sistema, tornando-o ainda mais adaptável e eficiente para enfrentar os desafios dinâmicos do comércio eletrônico.

Em síntese, o sistema de gerenciamento de pedidos demonstrou estar alinhado com as necessidades específicas de *marketplaces*, contribuindo para uma gestão ágil, segura e adaptada às nuances do dinâmico cenário do comércio eletrônico. O compromisso com a inovação e o uso de tecnologias modernas destacaram-se como pilares do sucesso na consecução dos objetivos propostos.

## REFERÊNCIAS

ALVES, Maria Bernadete Martins; ARRUDA, Susana Margareth. Como fazer referências: bibliográficas, eletrônicas e demais formas de documento. Florianópolis: Universidade Federal de Santa Catarina, Biblioteca Universitária, c2001. Documento não publicado.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 10520: informação e documentação: citações em documentos: apresentação. Rio de Janeiro: ABNT, 2023.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 14724: informação e documentação: trabalhos acadêmicos: apresentação. Rio de Janeiro: ABNT, 2011.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 6024: informação e documentação: numeração progressiva das seções de um documento escrito: apresentação. Rio de Janeiro: ABNT, 2012a.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 6027: informação e documentação – sumário – apresentação. Rio de Janeiro: ABNT, 2012b.

AWARI. MongoDB: o que é e como utilizar esse banco de dados?, 2022. Disponível em: <<https://awari.com.br/o-que-e-mongo-db/>>. Acesso em: 25 de nov. 2023. HOWS, David; MEMBREY, Peter; PLUGGE, Eelco. Introdução ao MongoDB. Novatec Editora, 2019.

BECKER, Annie. Ed. Electronic commerce: Concepts, Methodologies, Tools, and Applications: IGI global, 2007.

BOOTSTRAP. Build fast, responsive sites with Bootstrap. Disponível em: <https://getbootstrap.com/>. Acesso em: 29 nov. 2023.

BURATTO, Caio. O que é hospedagem de site? 2023. Disponível em: <https://www.locaweb.com.br/conteudos/o-que-e-hospedagem-de-site/>. Acesso em: 30 de nov. 2023.

CHEAH, I., PHAU, I., & LIANG, J. (2015). Factors influencing consumers' attitudes and purchase intentions of e-deals. *Marketing intelligence & planning*, 33(5), 763-783.

DURSKI, G. (2003). Avaliação do desempenho em cadeias de suprimentos. *Revista da FAE, Curitiba*, 6(1), 27-38

EBIT. (2018). Webshoppers. <https://www.ebit.com.br/webshoppers>

ELMASRI, Ramez; NAVATHE, Shamkant B. SISTEMAS DE BANCO DE DADOS. 6. ed. São Paulo: Addison Wesley, 2011.

FILIPOVA, Olga et al. Backend development. *Software Development From A to Z: A Deep Dive into all the Roles Involved in the Creation of Software*, p. 101-131, 2018.

FRANCO, R. de S., MAIA, L. C. de C., & FILHO, R. G. de S. (2020). Avaliação do Nível de Serviço ao Cliente no Contexto do Marketplace da Empresa Alfa. *Revista Ciências Administrativas*, 26(1). DOI: 10.5020/2318-0722.2020.26.1.9382

GONÇALVES, Edson. Desenvolvendo Aplicações web com NetBeans IDE 5.5. Editora Ciência Moderna, 2007.

GUISSONI, L. A., & FARINHA, R. L. (2019). E-commerce com resultado. *GV-executivo*, 18(1), 40-42.

HOSTINGER. Web Server: O que é e Como Funciona? Disponível em: <https://www.hostinger.com.br/tutoriais/o-que-e-web-server>. Acesso em: 30 de nov. 2023.

HUBSPOT. Entenda o que é Marketplace, como funciona e suas vantagens. Disponível em: <https://br.hubspot.com/blog/marketing/o-que-e-marketplace#:~:text=Marketplace%20%C3%A9%20uma%20esp%C3%A9cie%20de,das%20vendas%20para%20a%20plataforma>. Acesso em: 30/11/2023.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. Censo Demográfico. Rio de Janeiro, [2010]. Disponível em: <https://www.ibge.gov.br/estatisticas/sociais/educacao/9662-censo-demografico-2010.html?=&t=resultados>. Acesso em: 29 mar. 2022.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. Normas de apresentação tabular. Rio de Janeiro: IBGE, 1993. Disponível em: <http://biblioteca.ibge.gov.br/visualizacao/monografias/GEBIS%20-%20RJ/normastabular.pdf>. Acesso em: 20 jan. 2020.

KOHN, K., & MORAES, C. H. de. (2007). O impacto das novas tecnologias na sociedade: conceitos e características da Sociedade da Informação e da Sociedade Digital. Trabalho apresentado no XXX Congresso Brasileiro de Ciências da Comunicação, Santos, 29 de agosto a 2 de setembro de 2007.

LEUSCHNER, R., CHARVET, F., & ROGERS, D. S. (2013). A meta-analysis of logistics customer service. *Journal of supply chain management*, 49(1), 47-63.

LIN, Jimmy; EL GEBALY, Kareem. The future of big data is... javascript?. *IEEE Internet Computing*, v. 20, n. 5, p. 82-88, 2016.

MACHADO, Amanda. Qual a diferença entre front-end e back-end? 2022. Disponível em: <https://tecnoblog.net/responde/qual-a-diferenca-entre-front-end-e-back-end/#:~:text=Ou%20seja%2C%20o%20back%2Dend,menus%2C%20imagens%20entre%20o%20funcionalidades.>>. Acesso em: 21 de nov. 2023.

MOHAMAD, Mohaizi Bin et al. Enterprise Problems and Proposed Solutions Using the Concept of E-Commerce. In: 2021 2nd International Conference on Smart Computing and Electronic Enterprise (ICSCEE). IEEE, 2021. p. 186-192.

MOZILLA. HTTP. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Web/HTTP/>. Acesso em: 28 nov. 2023.

MURPHY, C. et al. *Beginning HTML5 and CSS3: The Web Evolved*. Apress, 2012.

Neto, Antenor. Melhores Marketplaces para vender: 18 Principais [2024]. Disponível em: <https://www.tray.com.br/escola/melhores-marketplaces-para-vender/>. Acesso em: 10 dez. 2023.

NOLETO, Cairo. Aplicações web: entenda o que são e como funcionam! 2020. Disponível em: <<https://blog.betrybe.com/desenvolvimento-web/aplicacoes-web/>>. Acesso em: 28 nov. 2023

NORONHA, Joseph Terence Peter; KHAN, Shariq Mahmood; SEBASTIAN, Stina; MUSTAFA, Syed Muhammad Nabeel. A study on the E-commerce Trends using Data Analysis. 2022 3Rd International Conference On Innovations In Computer Science & Software Engineering (Iconics), [S.L.], 14 dez. 2022. IEEE. <http://dx.doi.org/10.1109/iconics56716.2022.10100466>.

ORACLE. O que é um banco de dados. Disponível em: <https://www.oracle.com/br/database/what-is-database/>. Acesso em: 15 nov. 2023.

PAUTASSO, Cesare. A Brief History of Liquid Software. In: 2023 IEEE International Conference on Edge Computing and Communications (EDGE). Lugano, Suíça: [s.n.], 2023. DOI: 10.1109/EDGE60047.2023.00058.

SACCOL, Amarolinda Zanela; REINHARD, Nicolau. Tecnologias de informação móveis, sem fio e ubíquas: definições, estado-da-arte e oportunidades de pesquisa. RAC: Revista de Administração Contemporânea, v.11, n.4, 2007.

SANTOS, Daiane Rodrigues; BASTOS, Brena Ramalho; GABRIEL, Julia Barreto. Vendas no varejo eletrônico (via internet) no Brasil antes e depois da popularização dos smartphones. Brazilian Applied Science Review, v. 2, n. 5, p. 1566-1578, 2018.

SCHEIDT, Felipe Alex. Fundamentos de CSS: criando design para sistemas web. Foz do Iguaçu: Outbox Livros Digitais, 2015.

SELENIUM. O Projeto Selenium de Automação de Navegadores. Disponível em: <https://www.selenium.dev/pt-br/documentation/#:~:text=No%20n%C3%BAcleo%20do%20Selenium%20est%C3%A1,simples%20que%20voc%C3%AA%20pode%20fazer>. Acesso em: 29 nov. 2023.

SHOPIFY. Inventory Management Systems: Ultimate Buying Guide. Disponível em: <https://www.shopify.com/au/enterprise/inventory-management-system>. Acesso em: 10 de dez. 2023.

SIMÃO, L., & GONÇALVES, M. (2015). Sistemática para seleção de um portfólio de estratégias de postergação para cadeias suprimentos. GEPROS: Gestão da Produção, Operações e Sistemas, 10(1), 35-47.

SOMMERVILLE, I. Engenharia de Software. 9ª ed. São Paulo, SP: Pearson, 2011.

SOUZA, Ivan. Saiba o que é NPM (Node Package Manager) e como instalar, 2020. Disponível em: <<https://rockcontent.com/br/blog/npm/>>. Acesso em: 28 de nov. 2023.

TEIXEIRA, Tarcisio. Comércio Eletrônico: Conforme o Marco Civil da Internet e a regulamentação do e-commerce no Brasil. São Paulo: Saraiva, 2015.

TILKOV, Stefan; VINOSKI, Steve. Node.js: Using JavaScript to build high-performance network programs. IEEE Internet Computing, v. 14, n. 6, p. 80-83, 2010.

TRUECOMMERCE. Importance of Inventory Management in the Marketplace. Disponível em: <https://www.truecommerce.com/blog/importance-marketplace-inventory-management/>. Acesso em: 13 de dez. 2023.

VISUAL STUDIO CODE. Getting started, 2023. Disponível em: <https://code.visualstudio.com/docs>. Acesso em: 20 de nov. 2023.

WHATWG. HTML: Living Standard. Whatwg.org, 2023. <https://html.spec.whatwg.org/> Acesso em: 28 de nov. 2023.