**SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL - SENAI**

**“Henrique Lupo”**

Adriele Zavan Luiz

Ana Julia Santos Araujo

Evelyn Rocha

Isadora Dantas Melli

Octávio Augusto Freitas da Silva

**INVENT +**

**Sistema de gestão do almoxarifado**

**ARARAQUARA-SP**

**2024**

Adriele Zavan Luiz

Ana Julia Santos Araujo

Evelyn Rocha

Isadora Dantas Melli

Octávio Augusto Freitas da Silva

**INVENT +**

**Sistema de gestão do almoxarifado**

Trabalho de Conclusão de Curso apresentado ao instituto do Serviço Nacional De Aprendizagem Industrial – SENAI “Henrique Lupo”, como parte dos requisitos para a obtenção do título de Técnico em Desenvolvimento de Sistemas.

Orientadores:

Prof. Alex Stocco

Prof. Ivo C. Neto

**ARARAQUARA-SP**

**2024**

**AGRADECIMENTOS**

Agradecemos primeiramente a Deus, por conceder-nos vida e forças para superar todos os desafios ao longo da elaboração deste trabalho. Nossa profunda gratidão aos nossos pais, irmãos e namorados cujo apoio incondicional e compreensão durante o período foram fundamentais para a realização deste projeto.

Agradecemos também aos meus professores Alex Stocco e Ivo Conceição Neto, cujas correções e ensinamentos foram cruciais para nosso desenvolvimento profissional e acadêmico ao longo do curso. À instituição de ensino SESI e SENAI, que desempenhou um papel essencial na nossa formação, Agradecemos pela dedicação e pelas valiosas lições que nos foram proporcionadas ao longo dos anos. Outrossim gostaríamos de expressar nossa profunda gratidão à almoxarife Patrícia Lima, que desempenhou um papel fundamental no desenvolvimento do nosso Trabalho de Conclusão de Curso. Sua colaboração foi essencial para o sucesso deste projeto.

Finalmente, expressamos nossa gratidão a todos que, de alguma forma, contribuíram para a realização deste trabalho. Cada apoio e contribuição foram fundamentais para o sucesso deste projeto.

“Não podemos prever o futuro, mas podemos criá-lo.” - Peter Drucke

**RESUMO**

A presente pesquisa trata-se de um estudo sobre a implementação de um sistema de gestão para almoxarifados em ambientes educacionais, visando melhorar a eficiência operacional e solucionar desafios enfrentados pela administradora do almoxarifado. A pesquisa foi aplicada através de um estudo de caso na instituição Serviço Nacional De Aprendizagem Industrial – SENAI “Henrique Lupo”, onde foram analisadas as necessidades específicas do almoxarifado e o impacto da automação no controle de materiais. Esse estudo tem como objetivo geral desenvolver e implantar uma plataforma web que facilite a organização e o controle de materiais. O sistema permitirá que os professores façam reservas de materiais de forma direta e simples através de um site online. Os pedidos serão encaminhados para o administrador do almoxarifado, que poderá revisar, autorizar e controlar a retirada dos itens solicitados. Além disso, o sistema incluirá funcionalidades para monitorar o estoque e gerar alertas quando os níveis de materiais estiverem abaixo do limite estipulado, ajudando a prevenir a falta de itens essenciais e a garantir um abastecimento oportuno. Os métodos utilizados na pesquisa foram explicativos e descritivos, combinando análises qualitativas e quantitativas. A pesquisa constatou que a implementação do sistema proposto não só melhorou o controle e a organização dos materiais, mas também reduziu significativamente o tempo gasto na gestão de estoque, beneficiando tanto a administração quanto os usuários finais, como os professores.

**Palavras-chave:** Gestão; almoxarifado; automação; eficiência.

**ABSTRACT**

This research presents a study on the implementation of a warehouse management system in educational environments, aimed at enhancing operational efficiency and addressing challenges faced by warehouse administrators. The study was conducted through a case study at the Serviço Nacional De Aprendizagem Industrial – SENAI “Henrique Lupo,” where the specific needs of the warehouse were analyzed, and the impact of automation on material control was evaluated. The primary objective of this research is to develop and deploy a web-based platform that streamlines the organization and management of materials. The system will enable teachers to make material reservations directly and easily through an online site. Requests will be forwarded to the warehouse administrator, who will review, authorize, and manage the withdrawal of requested items. Additionally, the system will feature functionalities to monitor stock levels and generate alerts when material levels fall below the stipulated threshold, thereby preventing shortages and ensuring timely replenishment. The research employed explanatory and descriptive methods, combining qualitative and quantitative analyses. The findings indicate that the implementation of the proposed system significantly improved material control and organization, reduced time spent on inventory management, and provided benefits to both the administration and end-users, such as teachers.

**Keywords:** Management; warehouse; automation; efficiency.

**LISTA DE ILUSTRAÇÕES**

**LISTA DE QUADROS**

[Quadro 1 - Prioridade dos Requisitos 22](#_heading=h.2r0uhxc)

[Quadro 2 - RF001 – Cadastro 33](#_heading=h.1664s55)

[Quadro 3 – RF002 Sistema de Login Inicial 34](#_heading=h.3q5sasy)

[Quadro 4 - RF003 - Página Inicial 34](#_heading=h.28h4qwu)

[Quadro 5 - RF004 - Cadastro de Materiais 36](#_heading=h.37m2jsg)

[Quadro 6 - RF005 - Controle de Estoque 37](#_heading=h.46r0co2)

[Quadro 7 - RF006 - Requisições e Entregas 37](#_heading=h.111kx3o)

[Quadro 8 - RF007 – Relatório 38](#_heading=h.206ipza)

[Quadro 9 - RNF001 - Criptografia de dados sensíveis 39](#_heading=h.1egqt2p)

[Quadro 10 - RNF002 - Facilidade de Uso 40](#_heading=h.2dlolyb)

[Quadro 11 - REG001 - Criptografia 41](#_heading=h.1rvwp1q)

**LISTA DE ABREVIATURAS E SIGLAS**

SENAI - Serviço Nacional de Aprendizagem Industrial

HTML - HyperText Markup Language

CSS - Cascading Style Sheet

API –

JSON

**SUMÁRIO**

[1 INTRODUÇÃO 13](#_heading=h.25b2l0r)

[1.1 Objetivo Geral 14](#_heading=h.30j0zll)

[1.2 Objetivos Específicos 14](#_heading=h.1fob9te)

[1.3 Justificativa 15](#_heading=h.3znysh7)

[2 DESENVOLVIMENTO 17](#_heading=h.2et92p0)

[2.1 Pesquisa do tema 17](#_heading=h.tyjcwt)

[2.2 Entrevista 17](#_heading=h.3dy6vkm)

[2.2.1](#_heading=h.1t3h5sf) Gestão do almoxarifado 18

[2.2.2 Tecnologias e ferramentas utilizadas na gestão de almoxarifado 19](#_heading=h.4d34og8)

[2.2.3 Almoxarife 20](#_heading=h.2s8eyo1)

[2.2.4 Estudos de caso 21](#_heading=h.17dp8vu)

[2.3 Levantamento de Requisitos 22](#_heading=h.3rdcrjn)

[2.3.1 Tipos de Requisitos 22](#_heading=h.26in1rg)

[2.3.2 Requisitos Funcionais 22](#_heading=h.lnxbz9)

[2.3.3 Requisitos Não-Funcionais 22](#_heading=h.35nkun2)

[2.3.4 Regras de negócios 22](#_heading=h.1ksv4uv)

[2.3.5 Prioridade dos Requisitos 23](#_heading=h.44sinio)

[2.4 Métodos utilizados 23](#_heading=h.z337ya)

[2.4.1 HTML 24](#_heading=h.3j2qqm3)

[2.4.2 Backend 24](#_heading=h.1y810tw)

[2.4.3 CSS 24](#_heading=h.2xcytpi)

[2.4.4 Frontend 25](#_heading=h.1ci93xb)

[2.4.5 Python 25](#_heading=h.3whwml4)

[2.4.6 Flask 26](#_heading=h.2bn6wsx)

[2.4.7 API 26](#_heading=h.qsh70q)

[2.4.8 Java Script 27](#_heading=h.3as4poj)

[2.4.9 AJAX 27](#_heading=h.1pxezwc)

[2.4.10 Jsonify 27](#_heading=h.49x2ik5)

[2.4.11 JSON 27](#_heading=h.2p2csry)

[2.4.12 Bootstrap 28](#_heading=h.147n2zr)

[2.4.13 Flash 28](#_heading=h.3o7alnk)

[2.4.14 Banco de Dados 28](#_heading=h.23ckvvd)

[2.4.15 MySQL 29](#_heading=h.ihv636)

[2.4.16 Figma 29](#_heading=h.32hioqz)

[3 DESCRIÇÃO GERAL DO SISTEMA 31](#_heading=h.1hmsyys)

[3.1 Funcionalidades principais 32](#_heading=h.41mghml)

[3.2 Benefícios esperados 32](#_heading=h.2grqrue)

[3.3](#_heading=h.vx1227) Público-Alvo 32

[4](#_heading=h.3fwokq0) REQUISITOS FUNCIONAIS 33

[4.1 RF001 – Cadastro 33](#_heading=h.1v1yuxt)

[4.2 RF002 - Sistema de Login Inicial 33](#_heading=h.2u6wntf)

[4.3](#_heading=h.3tbugp1) RF003 - Página Inicial 34

[4.4](#_heading=h.nmf14n) RF004 – Cadastro de Materiais 36

[4.5](#_heading=h.1mrcu09) RF005 - Controle de Estoque 37

[4.6](#_heading=h.2lwamvv) RF006 - Requisições e Entregas 37

[4.7](#_heading=h.3l18frh) RF007 – Relatório 38

[5](#_heading=h.4k668n3) REQUISITOS NÃO-FUNCIONAIS 39

[5.1 RNF001 - Criptografia de dados sensíveis 39](#_heading=h.2zbgiuw)

[5.2 RNF002 - Facilidade de Uso 40](#_heading=h.3ygebqi)

[6 REGRAS DE NEGÓCIOS 41](#_heading=h.sqyw64)

[6.1 REG001 - Criptografia 41](#_heading=h.3cqmetx)

[REFERÊNCIAS 43](#_heading=h.4bvk7pj)

# 1 INTRODUÇÃO

O curso Técnico em Desenvolvimento de Sistemas visa preparar profissionais para criar e gerenciar soluções tecnológicas eficazes. Nesse sentindo, com um currículo diversificado, o curso abrange temas como fundamentos de programação, desenvolvimento de software, gerenciamento de banco de dados e análise de sistemas. Com esses conteúdos, os alunos ganham experiência prática em várias linguagens de programação e ferramentas de desenvolvimento, além de aprender sobre metodologias ágeis e práticas de segurança da informação. O objetivo é formar técnicos capacitados para desenvolver sistemas e aplicações que atendam às demandas e desafios do setor tecnológico.

Desse modo, optamos por desenvolver um sistema de gestão de almoxarifado. Esta escolha visa aplicar os conhecimentos adquiridos ao longo do curso em um projeto prático e relevante. O sistema tem como objetivo aprimorar a eficiência na administração de estoques e no controle de materiais, utilizando conceitos de programação. Com este projeto, busca-se criar uma solução eficaz para a gestão de recursos, que possa atender às necessidades de diversas organizações. Buscamos trazer soluções para o setor do almoxarifado da escola, pois este é responsável pelo armazenamento e controle de materiais essenciais para o funcionamento diário da instituição. Atualmente, o setor enfrenta desafios na organização e distribuição desses recursos, devido à falta de um sistema eficiente de gerenciamento, o que leva a atrasos no atendimento de solicitações e à dificuldade de prever a reposição de itens. Portanto, um sistema automatizado ajudará a otimizar esses processos e melhorar a eficiência operacional.

O sistema proposto é uma plataforma web que facilitará a organização e o controle de materiais, visando solucionar os desafios enfrentados pela administradora do almoxarifado e melhorar a eficiência operacional. O problema central é a dificuldade de gerenciar o estoque devido às constantes solicitações de materiais, que exigem que a administradora interrompa suas atividades para atender a essas demandas. A nova plataforma permitirá que os professores façam reservas de materiais de forma direta e simples através de um site online. Os pedidos serão encaminhados para a administradora do almoxarifado, que poderá revisar, autorizar e controlar a retirada dos itens solicitados. Além disso, o sistema incluirá uma funcionalidade para monitorar o estoque e gerar alertas quando os níveis de materiais estiverem abaixo do limite estipulado, ajudando a prevenir a falta de itens essenciais e a garantir um abastecimento oportuno.

## 1.1 Objetivo Geral

O objetivo geral desta pesquisa é desenvolver e implantar uma plataforma web para a gestão de almoxarifados em ambientes educacionais, com foco na melhoria da eficiência operacional e na solução de desafios enfrentados pela administradora do almoxarifado.

## 1.2 Objetivos Específicos

* Observar detalhadamente o ambiente de trabalho, identificando os principais

problemas, como a sobrecarga enfrentada pela responsável pela gestão do almoxarifado, que lida com um ritmo acelerado e muitas responsabilidades. Durante a visita, foi constatado que a assistente administrativa precisa dar conta de todas as tarefas sozinha, o que torna o trabalho exaustivo e dificulta a gestão eficiente do tempo. Essa análise tem como objetivo identificar melhorias para atribuir em nosso sistema.

* Coletar e analisar as informações recolhidas durante a visita ao almoxarifado

e a entrevista com a responsável pela gestão do almoxarifado, organizando de forma clara as principais dificuldades e desafios operacionais. Essa etapa busca documentar as necessidades específicas do ambiente, como a sobrecarga de trabalho, falta de tempo e falta de eficiência nos processos, para garantir que o sistema a ser desenvolvido atenda a essas demandas e ofereça soluções práticas que melhorem a gestão e o fluxo de trabalho.

* Avaliar e escolher as ferramentas tecnológicas mais adequadas para o

desenvolvimento do sistema, considerando critérios essenciais como a funcionalidade que cada tecnologia oferece. Essa etapa envolve a análise de diversas linguagens de programação, bibliotecas e frameworks. O objetivo é garantir que as escolhas feitas proporcionem um ambiente de desenvolvimento eficiente, que atenda às necessidades do projeto e possibilite futuras atualizações e melhorias.

* Criar uma representação visual que esquematize as telas e funcionalidades

do sistema, fornecendo uma visão geral da interface do usuário. Essa etapa envolve o desenho de diagramas que detalham a navegação entre seções, a disposição dos elementos e a organização das informações de forma intuitiva. O *wireframe* deve incluir menus, botões, campos de entrada e áreas de conteúdo, priorizando uma experiência de usuário fluida e acessível, servindo como base para as fases de design e desenvolvimento do sistema. Em conjunto, foi feito o estudo de cores com o intuito de definir uma paleta que otimize a experiência do usuário, proporcionando uma interface intuitiva e agradável.

* Implementar as funcionalidades de acordo com o planejamento, garantindo

que as necessidades identificadas sejam atendidas de maneira eficiente. Essa etapa envolve a codificação de recursos como controle de estoque, gestão de requisições, geração de relatórios e automação de processos. Testes contínuos são realizados para garantir a funcionalidade e otimizar a experiência do usuário. O objetivo é entregar uma solução integrada que melhore a eficiência operacional do almoxarifado e facilite o trabalho da equipe.

## 1.3 Justificativa

A gestão eficaz de almoxarifados é crucial para o bom funcionamento de instituições educacionais, onde a organização e o controle de materiais impactam diretamente a eficiência das operações e a qualidade do ensino. No contexto da instituição Serviço Nacional De Aprendizagem Industrial – SENAI “Henrique Lupo”, a administração do almoxarifado enfrenta desafios significativos devido aos processos manuais e à falta de um sistema integrado que possa facilitar a reserva e o controle dos materiais. Esses desafios incluem a alta demanda de tempo para a gestão de estoque, a possibilidade de erros humanos e a dificuldade em manter um controle preciso dos níveis de materiais.

A justificativa para o desenvolvimento deste sistema está fundamentada na necessidade de melhorar a eficiência operacional, reduzir erros e otimizar o tempo de gestão de estoque, resultando em benefícios significativos para a administração da instituição e para os usuários finais. Logo, a implementação de uma plataforma web para a gestão de almoxarifado surge como uma solução eficaz para resolver esses problemas.

A pesquisa busca não apenas resolver as questões atuais enfrentadas pela instituição, mas também fornece uma solução replicável que pode ser adaptada para outras instituições educacionais com desafios semelhantes. A modernização do processo de gestão de almoxarifado através da automação é essencial para atender às demandas crescentes e garantir a continuidade das operações de forma eficaz e eficiente.

# 2 DESENVOLVIMENTO

## 2.1 Pesquisa do Tema

Primeiramente o grupo, durante algumas semanas, desenvolveu pesquisas na busca de projetos procurando uma área relevante e que pudesse trazer melhorias significativas para a Escola SENAI. Inicialmente, estávamos em busca de um tema que se alinhasse com os objetivos de aprimorar a gestão e a eficiência dentro da instituição.

Durante as primeiras etapas do planejamento, conversamos com nossos professores para obtermos orientações e sugestões sobre possíveis temas. Esses diálogos foram fundamentais para afunilar as opções e garantir que o tema escolhido não apenas fosse interessante, mas também oferecesse um impacto positivo e prático para a escola. Após essas conversas, foi iniciado o processo de pesquisas de diferentes áreas dentro da Escola SENAI que pudessem beneficiar-se de uma solução tecnológica. Um dos temas que se destacou foi a necessidade de melhorar a organização e a gestão do almoxarifado da escola. Identificamos que, embora o almoxarifado desempenhasse um papel crucial no fornecimento de materiais e recursos, havia uma oportunidade significativa para otimizar seus processos.

O tema do sistema web para a organização do almoxarifado foi escolhido com base na análise das necessidades identificadas e na relevância do problema. A decisão de desenvolver um sistema web surgiu da constatação de que uma solução tecnológica poderia transformar a gestão do almoxarifado, tornando-a mais eficiente e organizada. A ideia é criar uma plataforma que facilite o controle de estoque, o gerenciamento de compras e o processo de empréstimo de materiais, além de melhorar a comunicação e a coordenação entre os envolvidos.

## 2.2 Entrevista

Antes da elaboração de requisitos, o grupo realizou uma visita ao almoxarifado da escola, tendo a oportunidade de conhecer o ambiente e entrevistar a almoxarife responsável pelo setor. Foi detalhado as principais responsabilidades e atividades que ela realiza no gerenciamento do almoxarifado. Ela explicou que o almoxarifado da escola SENAI é dividido em duas partes, sendo uma delas administrada pela sede, onde a almoxarife é responsável apenas pela compra de itens, e a outra parte, na qual ela é totalmente responsável pela gestão de itens de papelaria e uniformes. Isso inclui a compra de materiais de escritório, como papel e canetas, além da administração e distribuição dos uniformes da escola. Além disso, ela também é responsável pelo empréstimo de materiais para funcionários, mantendo um registro dos itens emprestados.

Por fim, a almoxarife explicou que o almoxarifado realiza a organização e o controle de estoque, garantindo que todos os materiais estejam devidamente armazenados e que o inventário esteja sempre atualizado.

### 2.2.1 Gestão do almoxarifado

A gestão de almoxarifado é essencial para garantir a eficiência operacional em qualquer instituição educacional, incluindo o SENAI. Entre os conceitos básicos desta área, destaca-se o controle de estoque, que envolve o monitoramento e gerenciamento dos materiais armazenados. Esse controle é fundamental para assegurar que a quantidade certa de itens esteja disponível quando necessário, prevenindo excessos ou faltas e minimizando o risco de obsolescência.

O registro de entrada e saída é um aspecto crucial, que consiste na documentação dos itens que chegam ao almoxarifado e daqueles que são retirados ou entregues. Manter registros precisos permite a geração de relatórios confiáveis, o planejamento de compras e a manutenção adequada dos níveis de estoque. Complementar a isso, o reabastecimento e a criação de pedidos para novos materiais são baseados na análise do consumo e previsões de demanda, assegurando que o almoxarifado não fique sem itens essenciais e otimizando a gestão de recursos financeiros.

A classificação e o armazenamento dos materiais envolvem organizar os itens de acordo com categorias e garantir que sejam armazenados de maneira eficiente. Essa organização facilita o acesso rápido e melhora a eficiência operacional, reduzindo o tempo gasto na busca de itens. Além disso, a realização de inventários regulares e auditorias periódicas é vital para verificar a precisão dos registros e identificar discrepâncias, garantindo um controle eficaz dos materiais e a melhoria contínua dos processos.

A gestão de fornecedores e compras abrange a seleção e manutenção das relações com fornecedores, além da negociação de contratos e acompanhamento do desempenho. Uma boa gestão neste aspecto assegura a aquisição de materiais de qualidade a preços competitivos, fortalecendo a cadeia de suprimentos. Por fim, o uso de tecnologia e automação é cada vez mais importante, pois sistemas automatizados e softwares de gestão melhoram a precisão, reduzem o trabalho manual e agilizam as operações.

A segurança e a manutenção do almoxarifado também são aspectos essenciais, garantindo a integridade dos materiais armazenados e promovendo um ambiente seguro para os funcionários. Em conjunto, esses conceitos básicos formam a base para uma gestão eficiente de almoxarifado, contribuindo para a operação bem-sucedida e a satisfação dos usuários finais.

### 2.2.2 Tecnologias e ferramentas utilizadas na gestão de almoxarifado

Na gestão atual de almoxarifado do SENAI, ferramentas manuais são utilizadas para organizar e controlar os materiais de forma eficiente. A principal forma de organização manual é feita por setores, onde os materiais são separados e gerenciados de acordo com sua função. Essa organização é feita exclusivamente para atender as demandas internas de funcionários e professores da instituição, garantindo que os recursos estejam disponíveis para as atividades educacionais.

A ferramenta utilizada é o Excel, uma ferramenta de planilha eletrônica da Microsoft, que desempenha um papel central no controle dos materiais. O processo de baixa de materiais é feito de forma manual nessa planilha, onde a responsável pelo almoxarifado atualiza os registros à medida que os itens são retirados ou devolvidos.

Para os materiais que pertencem à sede ou que precisam de controle centralizado, a gestão é feita através de um site, que facilita o monitoramento em um nível mais amplo. No entanto, o uso desse sistema é limitado a itens específicos, enquanto a maior parte da gestão ainda depende de planilhas de Excel para o controle local.

Essas práticas refletem um sistema de gestão que, embora funcional, possui limitações relacionadas à automação e à eficiência. O uso manual de planilhas exige esforço contínuo e atenção aos detalhes, o que pode resultar em maior probabilidade de erros e retrabalho, além de ser menos eficaz em comparação a sistemas mais automatizados e integrados.

### 2.2.3 Assistente de serviços administrativos

O almoxarife da escola SENAI desempenha um papel vital na gestão do almoxarifado, garantindo a eficiência na administração dos materiais e recursos. Suas principais responsabilidades incluem o recebimento e inspeção de materiais, onde ela verifica a quantidade e a qualidade dos itens recebidos e resolve quaisquer discrepâncias. Ela também é responsável pelo armazenamento e organização dos materiais, assegurando que estejam armazenados de forma segura e acessível. No controle de estoque, a almoxarife monitora os níveis de materiais, realiza contagens periódicas e solicita reposição quando necessário para evitar faltas.

Além disso, o almoxarife atende às solicitações de diferentes setores do SENAI, preparando e entregando os materiais de acordo com a demanda. Ela elabora relatórios sobre o estado do estoque, que são úteis para análise de custos e planejamento orçamentário. Outra função importante é garantir a conformidade com normas de segurança e regulamentações, mantendo o almoxarifado em condições seguras e organizadas. O almoxarife também deve comunicar-se com outros setores e fornecedores para coordenar a distribuição e resolver eventuais problemas.

### 2.2.4 Estudos de caso

No estudo de caso sobre a gestão do almoxarifado na instituição SENAI "Henrique Lupo", identificou-se uma série de desafios operacionais que impactam diretamente a eficiência e a organização dos materiais. A entrevistada apontou que o maior problema enfrentado é o controle de estoque, que, devido à natureza manual do processo, não permite que ela dedique tempo suficiente para montar e organizar o estoque de maneira eficaz. Com dois estoques a serem gerenciados, e nenhum deles inserido em um sistema automatizado, a administração do almoxarifado torna-se complexa e trabalhosa.

Outro desafio mencionado foi a necessidade de interromper constantemente o trabalho em andamento para atender solicitações de materiais de professores e funcionários. A dinâmica atual exige que, para cada solicitação, a pessoa responsável tenha que registrar manualmente a quantidade de materiais retirados em planilhas de Excel. Essa tarefa, embora funcione como controle básico, é ineficaz para lidar com o volume de solicitações e a diversidade de materiais gerenciados. Além disso, o fato de a comunicação com os professores ser feita verbalmente dificulta o acompanhamento do que está ou não disponível no estoque, o que leva a inconsistências e falhas no controle.

Um ponto crítico identificado foi a falta de visibilidade sobre os itens disponíveis. Em muitos casos, os professores fazem pedidos sem saber se o material está disponível ou não, o que gera confusões e atrasos. A falta de organização e automação no processo de estoque já resultou em prejuízos financeiros, como o descarte de mais de 50.000 reais em materiais vencidos, devido à compra excessiva e falta de controle adequado. Embora o sistema da sede da instituição seja mais organizado e eficiente, esse controle não se aplica ao estoque gerenciado localmente, que ainda depende de métodos manuais para entradas e saídas de materiais.

## 2.3 Levantamento de Requisitos

De forma simples, o levantamento de requisitos trata-se do processo de compreensão e identificação das necessidades que o cliente espera ser solucionado pelo sistema que será desenvolvido, definindo a função que o software vai desempenhar. (MARQUES, 2023)

### 2.3.1 Tipos de Requisitos

Os requisitos para o desenvolvimento do Sistema serão divididos neste documento em duas categorias: Funcionais e Não Funcionais (também conhecidos como Atributos de Qualidade).

### 2.3.2 Requisitos Funcionais

Um requisito funcional é uma funcionalidade específica que o sistema legado

deve ter, ou seja, uma ação que precisa ser capaz de realizar.

#### 2.3.2.1 RF001 – Cadastro

**Quadro 2 - RF001 – Cadastro**

|  |  |
| --- | --- |
| Descrição | O sistema deve oferecer um sistema de login que permita que, caso o cargo do usuário seja Almoxarifado, ele tenha acesso total à plataforma. Esse processo inclui um fluxo específico para o primeiro login de novos usuários, que precisam incluir código “sn”, cargo, e-mail, senha para logar. |
| Pré-condições | O usuário deve ter um perfil válido e estar registrado no sistema. |
| Entradas | Login, senha, cargo. |
| Saídas | Acesso ao sistema conforme o perfil do usuário (admin ou solicitante de materiais). |
| Pós-Condições | O sistema autentica o usuário e o redireciona para a página inicial correspondente ao seu perfil. |
| Prioridade | Essencial |
| Dependência | Número de série (SN) armazenado no banco de dados, vinculado ao perfil do usuário. |

Fonte: Autoria própria (2024)

#### 2.3.2.2 RF002 - Sistema de Login Inicial

**Quadro 3 – RF002 Sistema de Login Inicial**

|  |  |
| --- | --- |
| Descrição | O sistema deve oferecer um sistema de login que permita que, caso o cargo do usuário seja Almoxarifado, ele tenha acesso total à plataforma. Esse processo inclui um fluxo específico para o primeiro login de novos usuários, que precisam incluir código “sn”, cargo, e-mail, senha para logar. |
| Pré-condições | O usuário deve ter um perfil válido e estar registrado no sistema. |
| Entradas | Login, senha, cargo. |
| Saídas | Acesso ao sistema conforme o perfil do usuário (admin ou solicitante de materiais). |
| Pós-Condições | O sistema autentica o usuário e o redireciona para a página inicial correspondente ao seu perfil. |
| Prioridade | Essencial |
| Dependência | Número de série (SN) armazenado no banco de dados, vinculado ao perfil do usuário. |

Fonte: Autoria própria (2024)

#### 2.3.2.3 RF003 - Página Inicial

**Quadro 4 - RF003 - Página Inicial**

|  |  |
| --- | --- |
| Descrição | A página inicial do sistema de almoxarifado serve como um dashboard central para monitorar e gerenciar as operações diárias de estoque, como devoluções, empréstimos, controle de estoque mínimo e saídas. Ela fornece uma visão geral rápida das atividades e status do inventário, permitindo aos usuários acessar diferentes seções do sistema com facilidade. |
| Pré-condições | O usuário deve estar autenticado no sistema. O usuário deve ter permissões adequadas para visualizar a página inicial e realizar ações relacionadas ao gerenciamento do almoxarifado. O sistema deve estar integrado com o banco de dados para obter as informações em tempo real. |
| Entradas | Ações do usuário, como clicar em botões para navegar entre as seções (Devoluções, Empréstimos, Estoque Mínimo, Saídas). Informações sobre o estado atual do estoque, como itens em devolução, empréstimos ativos, itens abaixo do estoque mínimo e saídas pendentes. |
| Saídas | Exibição de indicadores visuais sobre o status das operações (exemplo: quantidades de devoluções, itens abaixo do estoque mínimo, saídas). Atualização em tempo real do status dos itens e operações de estoque. Navegação para outras páginas do sistema, como "Produtos", "Entradas", "Saídas" e “Devoluções”. |
| Pós-Condições | O sistema deve refletir as ações realizadas pelo usuário, como a visualização de detalhes ou a atualização de informações. O usuário deve ser capaz de acessar rapidamente os módulos desejados para executar operações adicionais. O sistema deve continuar monitorando e atualizando os indicadores de status conforme novas transações ocorrem. |
| Prioridade | Essencial |
| Dependência | Usuário administrador, Banco de dados do inventário, Módulos de gerenciamento de estoque. |

Fonte: Autoria própria (2024)

#### 2.3.2.4 RF004 – Cadastro de Materiais

**Quadro 5 - RF004 - Cadastro de Materiais**

|  |  |
| --- | --- |
| Descrição | O sistema deve permitir o cadastro de novos materiais no almoxarifado com informações detalhadas, incluindo nome do produto, descrição, categoria, estoque mínimo e máximo. |
| Pré-condições | O usuário deve estar logado no sistema com um perfil de administrador. |
| Entradas | Descrição detalhada, categoria, estoque mínimo e máximo. |
| Saídas | Mensagem de confirmação do material no sistema e visualização do material cadastrado na lista de materiais. |
| Pós-Condições | As informações do material são armazenadas no banco de dados e visíveis para os administradores. |
| Prioridade | Essencial |
| Dependência | Usuário administrador. |

Fonte: Autoria própria (2024)

#### 2.3.2.5 RF005 - Controle de Estoque

**Quadro 6 - RF005 - Controle de Estoque**

|  |  |
| --- | --- |
| Descrição | O sistema deve gerenciar o controle de estoque, registrando entradas e saídas de produtos, atualizando automaticamente os níveis de estoque e gerando alertas quando os níveis atingem o estoque mínimo. |
| Pré-condições | O usuário deve estar logado e ter permissões adequadas para registrar entradas e saídas. |
| Entradas | Tipo de transação (entrada ou saída), quantidade, ID do material. |
| Saídas | Atualização dos níveis de estoque, alertas de estoque mínimo e máximo, e relatórios de inventário. |
| Pós-Condições | Os níveis de estoque são atualizados e alertas são enviados conforme necessário. |
| Prioridade | Essencial |
| Dependência | Usuário administrador. |

Fonte: Autoria própria (2024)

#### 2.6.6 RF006 - Requisições e Entregas

**Quadro 7 - RF006 - Requisições e Entregas**

|  |  |
| --- | --- |
| Descrição | O sistema deve permitir a criação de solicitações aprovação de requisições de materiais pelos usuários e o gerenciamento das entregas e materiais emprestados. |
| Pré-condições | Usuários devem estar logados e ter permissões adequadas para criar requisições. |
| Entradas | ID do material, quantidade solicitada, justificativa, status da requisição (aprovada ou rejeitada). |
| Saídas | Confirmação de requisição, status da requisição, registro de entrega de materiais, controle de materiais emprestados. |
| Pós-Condições | Requisições são registradas, e o status é atualizado conforme as ações tomadas: aprovada, pendente e negada. |
| Prioridade | Importante |
| Dependência | Usuário logado. |

Fonte: Autoria própria (2024)

#### 2.6.7 RF007 – Relatório

**Quadro 8 - RF007 – Relatório**

|  |  |
| --- | --- |
| Descrição | O sistema deve gerar relatórios e análises sobre o estoque e movimentação de materiais, com gráficos mensais. |
| Pré-condições | Os dados de movimentação e estoque devem estar atualizados. |
| Entradas | Período para o relatório (mensal, anual), tipo de análise (estoque, movimentação). |
| Saídas | Relatórios e gráficos detalhados sobre o estoque e movimentação de materiais. |
| Pós-Condições | Relatórios são gerados e podem ser visualizados ou exportados pelos usuários autorizados. |
| Prioridade | Essencial |
| Dependência | Usuário administrador para visualizar o relatório. |

Fonte: Autoria própria (2024)

#### 2.6.8 RF008 – Requisição de Materiais

**Quadro 9 - RF008 – Requisição de Materiais**

|  |  |
| --- | --- |
| Descrição | O sistema deve permitir que o usuário crie uma requisição de material, onde ele pode selecionar produtos, definir a quantidade e visualizar as requisições pendentes após o envio da solicitação. |
| Pré-condições | O usuário deve estar autenticado no sistema.  O catálogo de materiais/produtos deve estar disponível e atualizado. |
| Entradas | Seleção de produto(s) a ser requisitado.  Quantidade de cada produto solicitado. |
| Saídas | Relatórios e gráficos detalhados sobre o estoque e movimentação de materiais. |
| Pós-Condições | Confirmação da requisição de material.  Exibição de resumo da requisição criada (produtos e quantidades).  Exibição de todas as requisições pendentes do usuário, com informações como: código da requisição, data da solicitação, produtos e quantidades, e status (pendente, em processamento, entregue). |
| Prioridade | Essencial |
| Dependência | Cadastro e atualização de produtos.  Módulo de autenticação de usuários.  Função de criação e acompanhamento de requisições. |

Fonte: Autoria própria (2024)

#### 2.6.9 RF009 – Acessibilidade

**Quadro 10 - RF009 - Acessibilidade**

|  |  |
| --- | --- |
| Descrição | O sistema deve permitir ao usuário alternar entre o Modo Claro e o Modo Escuro para ajustar a aparência da interface conforme a preferência de visualização. |
| Pré-condições | O usuário deve estar autenticado ou utilizando o sistema.  O sistema deve ter os estilos de interface tanto para o Modo Claro quanto para o Modo Escuro implementados. |
| Entradas | Ação do usuário para selecionar o modo de visualização desejado (Claro ou Escuro), via botão dedicado. |
| Saídas | Atualização imediata da interface para o Modo Claro ou Modo Escuro, conforme a escolha do usuário. |
| Pós-Condições |  |
| Prioridade | Desejável |
| Dependência | Implementação dos temas visuais para Modo Claro e Modo Escuro |

Fonte: Autoria própria (2024)

### 2.3.3 Requisitos Não-Funcionais

Um requisito não funcional refere-se a características ou qualidades do sistema, como desempenho, segurança, facilidade de uso, confiabilidade, e assim por diante.

#### 2.3.3.1 RNF001 - Criptografia de dados sensíveis

**Quadro 11 – RNF001 - Criptografia de dados sensíveis**

|  |  |
| --- | --- |
| Descrição | As informações sensíveis e dados pessoais devem ser armazenados de forma criptografada e protegidos por autenticação multifatorial. |
| Critérios de Aceitação | Utilizar um algoritmo de criptografia forte e amplamente utilizado, como SHA-256 ou *bcrypt*.  As senhas criptografadas devem ser armazenadas separadamente dos demais dados do usuário, em uma tabela específica do banco de dados. |
| Impacto e observações | **Desempenho:** A criptografia pode  adicionar um pequeno overhead de processamento durante o cadastro e login de usuários.  **Complexidade:** Requer conhecimento técnico para implementar corretamente a criptografia. |
| Prioridade | importante |
| Dependência | Usuário administrador para visualizar o relatório. |

Fonte: Autoria própria (2024)

#### 2.3.3.2 RNF002 - Facilidade de Uso

**Quadro 12 - RNF002 - Facilidade de Uso**

|  |  |
| --- | --- |
| Descrição | O sistema deve ser projetado e desenvolvido de forma a ser intuitivo e fácil de usar por todos os usuários, independentemente do seu nível de conhecimento técnico. |
| Critérios de Aceitação | **Interface intuitiva:** A interface gráfica do usuário (GUI) deve ser clara, organizada e seguir padrões de design conhecidos (como material design ou human interface guidelines).  **Feedback visual**: O sistema deve fornecer feedback visual claro para as ações do usuário, como mensagens de confirmação, alertas e indicadores de progresso.  **Atalhos de teclado**: Implementar atalhos de teclado para as funções mais utilizadas, agilizando o trabalho dos usuários.  **Personalização**: Permitir que os usuários personalizem a interface de acordo com suas preferências.  **Treinamento**: Deve ser possível realizar o treinamento do sistema em menos de 48h. |
| Impacto e observações | **Complexidade:** Requer um técnico especializado em front end. |
| Prioridade | Importante |
| Dependência |  |

Fonte: Autoria própria (2024)

### 2.3.4 Regras de Negócios

Regras de negócios são, portanto, estratégias corporativas que devem ser

pensadas e executadas a partir da identidade da empresa, seus valores, missões e

planejamento estratégico.

#### 2.3.4.1 REG001 - Criptografia

**Quadro 13 - REG001 - Criptografia**

|  |  |
| --- | --- |
| Descrição | Todas as informações sensíveis armazenadas e transmitidas pelo Sistema de Gestão do Almoxarifado devem ser criptografadas para garantir a segurança e a proteção dos dados contra acessos não autorizados. |
| Observações | A criptografia é essencial para proteger informações confidenciais, como credenciais de login e dados pessoais, e garantir a integridade e a confidencialidade das informações trocadas entre o sistema e seus usuários. Isso ajuda a prevenir vazamentos de dados e a proteger a privacidade dos usuários e a integridade das informações no sistema. |
| Ações | Todas as informações sensíveis armazenadas no banco de dados, como senhas de usuários, informações de fornecedores e dados de movimentação, devem ser criptografadas usando algoritmos de criptografia robustos (por exemplo, AES-256).  As chaves de criptografia devem ser armazenadas separadamente do banco de dados e devem ser protegidas adequadamente.  As senhas dos usuários devem  ser criptografadas usando uma função de hash segura antes de serem armazenadas no banco de dados.  O sistema não deve armazenar senhas em texto claro nem permitir a recuperação direta das senhas. Em vez disso, deve implementar um mecanismo de redefinição de senha.  O sistema deve garantir que apenas usuários autenticados e autorizados possam acessar informações sensíveis. |

Fonte: Autoria própria (2024)

### 2.3.5 Prioridade dos Requisitos

Para estabelecer a prioridade dos requisitos, foram adotadas as denominações: essencial, importante e desejável, conforme descrição abaixo:

**Quadro 1 - Prioridade dos Requisitos**

|  |  |
| --- | --- |
| Essencial | São os requisitos fundamentais que devem ser atendidos para que o projeto funcione. Sem eles, o projeto não pode  ser concluído ou utilizado. |
| Importante | São requisitos que, embora não sejam críticos para o funcionamento básico, são muito desejáveis e melhoram significativamente a usabilidade e a eficácia do projeto. |
| Desejável | São requisitos que seriam ótimos de ter e podem adicionar valor extra, mas não são indispensáveis. Podem ser incluídos se houver tempo e recursos disponíveis. |

Fonte: Autoria própria (2024)

## 2.4 Métodos Utilizados

No desenvolvimento deste sistema, foram aplicadas diversas tecnologias que, em conjunto, viabilizam a criação de uma plataforma web funcional e interativa. HTML (*Hypertext Markup Language*) e CSS (*Cascading Style Sheets*) foram utilizados para estruturar e estilizar a interface do usuário, garantindo uma apresentação visual clara e atraente. Python, junto ao *framework* Flask, foi responsável pela construção do backend, possibilitando o tratamento das requisições e a implementação da lógica de negócios. O uso de JavaScript, juntamente com AJAX (*Asynchronous JavaScript and XML*), permitiu a criação de interações dinâmicas na página, sem a necessidade de recarregamento completo do conteúdo. A biblioteca jsonify foi empregada para converter dados em formato JSON, assegurando a comunicação eficaz entre frontend e backend. Bootstrap foi utilizado para criar uma interface responsiva e otimizada para diferentes dispositivos. Flash foi aplicado para a geração de mensagens temporárias no sistema, melhorando a experiência do usuário.

### 2.4.1 HTML

HTML ou Linguagem de Marcação de Hipertexto, é a base da criação e estruturação de páginas na web. Ele define a organização e disposição dos elementos, como textos, imagens, links, tabelas, vídeos e outros conteúdos que são exibidos nos navegadores. Utiliza uma série de tags (marcadores) que envolvem o conteúdo e atribuem funções a cada elemento na página.

Utilizando HTML, é possível estruturar formulários para cadastro de itens, tabelas para exibição dos produtos em estoque, e links para navegação entre diferentes funcionalidades, como consulta de itens, adição de novos produtos ou retirada de materiais.

### 2.4.2 Backend

Backend é o desenvolvimento de recursos e estrutura que garantem o funcionamento das ações realizadas pelos usuários em uma aplicação. Como o próprio nome sugere, o backend compreende tudo o que ocorre "por trás" de uma aplicação, sendo responsável pelo processamento de dados, regras de negócio e comunicação com o banco de dados. Ele gerencia a lógica do sistema, armazenamento de dados, autenticação de usuários e a manipulação das requisições recebidas do frontend (a interface com a qual os usuários interagem).

Desempenha um papel crucial ao lidar com o processamento de operações como o controle de estoque, registro de entradas e saídas de materiais, e geração de relatórios. Ele se conecta ao banco de dados para armazenar informações de produtos, fornecedores, e movimentações, garantindo que os dados estejam sempre atualizados e acessíveis. Além disso, cuida da autenticação de usuários, assegurando que apenas pessoas autorizadas tenham acesso ao sistema e às funcionalidades específicas, como a gestão de permissões de usuários e a realização de auditorias.

### 2.4.3 CSS

CSS (*Cascading Style Sheets*) é uma linguagem usada para estilizar e formatar elementos escritos em uma linguagem de marcação, como HTML. Ele permite separar o conteúdo da parte visual de um site, o que torna a manutenção muito mais fácil e possibilita a reutilização dos estilos em várias páginas. Assim, você pode atualizar o visual sem precisar mexer no conteúdo, tornando tudo mais prático e eficiente. Com o CSS, é possível controlar a aparência de uma página web, definindo cores, fontes, espaçamentos, margens, e o layout dos elementos. Pense no CSS como a "decoração" da sua página, permitindo ajustar a cor do texto e do fundo, o espaçamento entre parágrafos, e até criar layouts variados e responsivos que se adaptam a diferentes tamanhos de tela, como desktops, tablets e smartphones. Isso resulta em uma experiência de usuário mais consistente e agradável.

O CSS é essencial para tornar a navegação mais fácil e agradável para o usuário. Ele permite que o site se adapte a diferentes dispositivos, garantindo que a interface funcione bem tanto em desktops quanto em dispositivos móveis.

### 2.4.4 Frontend

Frontend é a parte de um site ou aplicativo com a qual os usuários interagem diretamente. Isso inclui elementos visuais como botões, menus, textos e imagens. Os principais componentes do frontend são HTML, que define a estrutura da página; CSS, que controla a apresentação visual; e JavaScript, que adiciona interatividade.

Basicamente, o frontend envolve a forma como as pessoas visualmente interagem e navegam pelo sistema. É essencial que os desenvolvedores se concentrem em como os usuários se sentem ao usar o sistema, garantindo que a navegação seja simples e prazerosa. O trabalho de frontend é fundamental para criar uma experiência intuitiva e envolvente.

### 2.4.5 Python

Python é uma linguagem de programação de alto nível, famosa por sua simplicidade e clareza, o que facilita tanto o aprendizado quanto a manutenção do código. Criada em 1991 por Guido van Rossum, ela se destacou rapidamente por sua versatilidade, sendo amplamente usada em diversas áreas, como inteligência artificial, desenvolvimento de sites, automação e testes de software.

Com uma biblioteca padrão extensa e uma comunidade ativa, Python oferece inúmeras ferramentas e suporte para quem a utiliza. Por ser uma linguagem de propósito geral, ele não se limita a resolver apenas um tipo de problema, tornando uma escolha perfeita para uma ampla variedade de projetos. Além disso, Python é compatível com diversos sistemas operacionais, como Linux, Mac e Windows, o que contribui ainda mais para sua popularidade entre desenvolvedores de todo o mundo.

Python foi utilizado para desenvolver a lógica de backend, facilitando o gerenciamento produtivo de dados e a execução de operações essenciais do sistema, facilitando o processamento de informações e o controle de atividades do sistema de uma maneira mais organizada, além de realizar operações complexas, como geração de relatórios e análises de estoque, permitindo que a equipe desenvolvesse novas funcionalidades e pudessem manter o código de forma simples.

### 2.4.6 Flask

Flask é um *microframework* para Python utilizado no desenvolvimento de aplicações web. Ele é leve e flexível, permitindo que os desenvolvedores criem aplicações de forma rápida e simples, com ferramentas essenciais como a navegação de URLs e o controle de requisições.

Ideal para protótipos e sites dinâmicos, Flask é amplamente utilizado por sua simplicidade e personalização, oferecendo liberdade para trabalhar com outras bibliotecas conforme a necessidade. Além disso, é uma alternativa gratuita e versátil, adequada para servidores com poucos recursos, tornando-o uma excelente escolha para programadores de todos os níveis.

A simplicidade do Flask permitiu que a equipe configurasse rapidamente as rotas para funcionalidades essenciais, como cadastro de produtos, consulta de estoque e registro de movimentações. Com suporte para extensões, é possível adicionar funcionalidades adicionais, como autenticação de usuários, conexão com bancos de dados e suporte a formulários, conforme necessário. Essa flexibilidade não só acelera o desenvolvimento, mas também possibilita a criação de um sistema adaptável, que pode crescer à medida que as demandas do negócio aumentam.

### 2.4.7 API

API (Application Programming Interface) é um aplicativo que permite a comunicação entre dois elementos de software por meio de um conjunto de definições e protocolos. Um exemplo é a aplicação de previsão do tempo no telefone, que se conecta ao sistema de dados meteorológicos do instituto meteorológico por meio de APIs para exibir atualizações diárias. Essas interfaces oferecem um conjunto de padrões que facilitam o desenvolvimento de plataformas de maneira mais simples e prática para os desenvolvedores, permitindo a criação de softwares, aplicativos, programas e diversas outras soluções.

### 2.4.8 JavaScript

JavaScript é uma linguagem de programação essencial para criar páginas da web interativas e dinâmicas. Ela permite adicionar funcionalidades como validação de formulários, animações, menus suspensos e gráficos interativos. O JavaScript funciona diretamente no navegador, o que deixa as páginas mais rápidas e melhora a experiência do usuário, tornando a navegação mais fluida e agradável. Além de ser extremamente importante para o desenvolvimento web, ele também pode ser usado tanto no frontend quanto no backend, além de dispositivos móveis ou fixos.

O JavaScript foi usado para implementar validações de formulários em tempo real, garantindo que os dados inseridos pelos usuários sejam corretos antes de serem enviados ao servidor. Através de requisições assíncronas, como AJAX, é possível atualizar partes da página sem precisar recarregá-la completamente, o que melhora a performance e a experiência geral. Também foi utilizado para melhorar a experiência do usuário ao interagir com o sistema.

### 2.4.9 AJAX

AJAX é uma técnica de desenvolvimento web que melhora a interatividade e a experiência do usuário ao permitir a atualização de partes específicas de uma página web sem recarregar a página inteira. Com o uso do AJAX, é possível trocar dados com o servidor sem interromper o que o usuário está fazendo, já que a página segue funcionando normalmente sem interrupções.

No projeto, o AJAX foi utilizado para facilitar a comunicação entre o front-end e as rotas no servidor Flask. Em vez de recarregar a página a cada interação do usuário, o AJAX permite que o indivíduo envie e receba dados de forma assíncrona, agilizando o processo.

### 2.4.10 Jsonify

Jsonifyé uma função do Flask, um microframework web em Python, usada para converter dados em formato JSON (JavaScript Object Notation) de forma simples. Geralmente é utilizada em APIs para retornar respostas em JSON, facilitando a troca de dados entre servidores e clientes na web.

### 2.4.11 JSON

JSON é uma formatação de troca de dados, que é fácil de ler, escrever, de interpretar e gerar para máquinas. Baseado em uma parte da linguagem de programação JavaScript, o formato JSON permite que objetos JavaScript sejam representados de forma comum para várias linguagens.

Ele foi utilizado no projeto para facilitar a visualização das informações em Python. Com estrutura fácil, ajuda na organização de dados, tornando a comunicação entre servidor e cliente mais acessível.

### 2.4.12 Bootstrap

O Bootstrap é um framework Frontend que permite acesso público a códigos colaborando na criação e desenvolvimento de sites, utilizado normalmente por desenvolvedores web. De forma gratuita, simplifica muitas etapas no desenvolvimento web, oferecendo elementos pré-estilizados, como botões e formulários, além de auxiliar na criação de layouts que se ajustam a diferentes tamanhos de tela. Ele foi utilizado como base durante toda a criação do Backend.

### 2.4.13 Flash

Flash é um framework para desenvolvimento de sistemas web em Python. Ele permite a criação rápida de sites e APIs, oferecendo recursos básicos, como roteamento de URLs e gerenciamento de requisições.

O Flash foi utilizado para exibir avisos importantes, como notificações de saída de conta, assim, sempre que o usuário fizer logout ou sua sessão expirar, o Flash será acionado para mostrar uma mensagem.

### 2.4.14 Banco de Dados

O banco de dados é uma biblioteca de informações que pode ser acessada facilmente, gerenciada e atualizada. Ele armazena dados de forma estruturada, permitindo que usuários e programas consultem e manipulem essas informações de maneira prática. No projeto, o banco de dados armazena todas as informações dos usuários e de materiais.

### 2.4.15 MySQL

MySQL é um sistema de gerenciamento de banco de dados relacional de código aberto, apoiado pela Oracle e baseado na linguagem SQL. Ele funciona em várias plataformas, como Linux, UNIX e Windows, e é frequentemente usado em aplicações da web.

O MySQL ajuda a armazenar e organizar informações de forma estruturada, como um grande caderno. Por exemplo, você pode ter uma lista de contatos com nomes, telefones e endereços. Quando precisa encontrar uma informação específica, como o telefone de um amigo, o MySQL permite que você busque rapidamente.

É amplamente utilizado por empresas para gerenciar dados importantes, como registros de clientes e produtos. Além disso, é gratuito e tem uma grande comunidade, oferecendo muitos recursos e suporte. Em resumo, MySQL facilita o trabalho com informações, tornando-o mais eficiente.

Com o MySQL, é possível estruturar tabelas que contêm informações detalhadas sobre cada item no estoque, facilitando consultas rápidas e precisas, como a verificação de níveis de estoque, produtos próximos ao vencimento ou movimentações recentes.

### 2.4.16 Figma

O Figma é uma plataforma colaborativa online, lançada em 2016 por Dylan Field e Evan Wallace, pertencente à empresa Figma, Inc. Seu objetivo é permitir a construção de interfaces, wireframes e protótipos, promovendo a colaboração entre pessoas e equipes. A ferramenta facilita o trabalho remoto e simultâneo de times de design, possibilitando a criação de produtos para diversas plataformas, mantendo a acessibilidade e oferecendo recursos para o design de telas de aplicações variadas.

#### 2.4.16.1 Wireframe

*Wireframe* é uma representação visual simples de um site ou aplicativo, mostrando a estrutura e a disposição de elementos como menus e botões, sem detalhes estéticos. Ele auxilia designers e desenvolvedores a planejar a interface, facilitando a comunicação de ideias e a identificação de problemas antes do desenvolvimento. Wireframes podem ser feitos em papel ou com ferramentas digitais e são uma etapa importante no design.

#### 2.4.16.2 Estudo de Cores

O estudo de cores é o processo de escolher e combinar tons, pensando nas sensações e mensagens que deseja transmitir. Foi importante para a equipe fazer o estudo de cores para criar uma identidade visual do sistema, influenciando na estética e usabilidade, destacando ações para o incentivo da interação, identificando cores eficazes, e pensando sempre na experiência do usuário.

## 2.5 Descrição Geral Do Sistema

O sistema proposto para a gestão de almoxarifado é uma plataforma web que visa automatizar e otimizar o controle e organização de materiais, com foco em ambientes educacionais como o SENAI. O objetivo principal do sistema é simplificar a administração do estoque, proporcionando uma interface intuitiva que permita a interação eficiente entre os usuários (professores e funcionários) e a administradora do almoxarifado.

Através da plataforma, os professores poderão realizar reservas de materiais diretamente, acessando uma lista atualizada de itens disponíveis no almoxarifado. O sistema exibirá o estoque em tempo real, facilitando a visualização de quais materiais estão disponíveis, eliminando a necessidade de solicitações verbais e de dependência de planilhas manuais. Essa funcionalidade reduz a interrupção do trabalho da administradora, que atualmente precisa parar suas atividades para atender a essas demandas.

Além disso, o sistema permitirá que a administradora do almoxarifado revise, autorize e controle os pedidos de forma mais organizada. Cada solicitação será registrada no sistema, com histórico detalhado de entradas e saídas de materiais. A plataforma também enviará alertas automáticos quando os níveis de estoque estiverem baixos, ajudando a evitar a falta de itens essenciais e promovendo o reabastecimento oportuno.

Uma funcionalidade crucial do sistema será o controle centralizado do estoque, que possibilitará a integração entre diferentes estoques, como os locais e os da sede. Isso garantirá uma visão ampla e organizada do inventário, minimizando perdas e excessos, e garantindo que todos os itens sejam utilizados dentro de seu prazo de validade.

Com essa plataforma, a expectativa é de melhorar significativamente a eficiência operacional, reduzindo o tempo dedicado à gestão manual de materiais e minimizando erros comuns em processos manuais. O sistema também permitirá a geração de relatórios para análise do consumo de materiais, auxiliando no planejamento estratégico do almoxarifado.

### 2.5.1 Funcionalidades principais

### 2.5.2 Benefícios esperados

### 2.5.3 Público-Alvo

* **Professores**: Usuários que precisam reservar e utilizar materiais para suas atividades acadêmicas.
* **Responsável** **pelo** **Almoxarifado**: Principal usuário que receberá e processará as solicitações de materiais e gerenciará o estoque.
* **SENAI** **em** **Geral**: Entidade que supervisionará a implementação e uso do sistema, garantindo que atenda às necessidades institucionais e melhore a eficiência operacional do almoxarifado.

## 

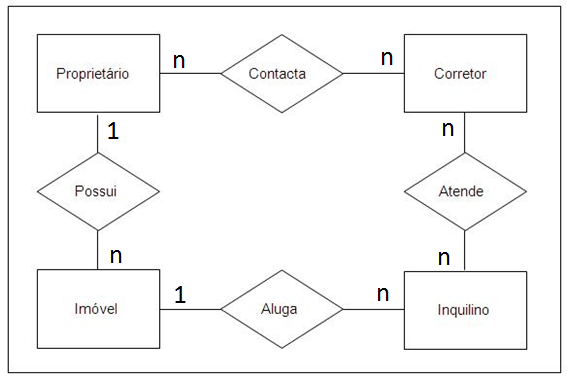
## 2.9 Diagrama

Um diagrama é uma representação visual simplificada, mostrando a aparência, estrutura ou funcionamento de algo. Normalmente é formado por palavras-chave ou conceitos que são ligados por linhas e setas que definem o raciocínio ou fluxo de ideias. Essa estrutura facilita o entendimento do tema, guiando o raciocínio de forma clara e organizada.

### 2.9.1 Diagrama MER

### 2.9.2 Diagrama DER

O diagrama DER (Diagrama Entidade-Relacionamento) é uma ferramenta no modelamento de banco de dados, é uma representação visual que se relacionam entre si dentro de um sistema de forma organizada e que atenda aos requisitos do projeto.

**Figura 1 - Diagrama Entidade Relacionamento de sistema de imobiliária**

Fonte:

# REFERÊNCIAS

AMOASEI, J. O que são regras de negócio? **Alura**, 2023. Disponivel em: <https://www.alura.com.br/artigos/o-que-sao-regras-de-negocio>. Acesso em: 12 Setembro 2024.

AWARI. Aprenda a Programar Em Python com Flash: o Guia Completo para Dominar a Linguagem de Programação Mais Popular do Momento. **Awari**, 2023. Disponivel em: <https://awari.com.br/aprenda-a-programar-em-python-com-flash-o-guia-completo-para-dominar-a-linguagem-de-programacao-mais-popular-do-momento/>. Acesso em: 19 Setembro 2024.

AWS AMAZON. O que é uma API (interface de programação de aplicações)? **AWS Amazon**. Disponivel em: <https://aws.amazon.com/pt/what-is/api/>. Acesso em: 19 Setembro 2024.

AWS AMAZON. Qual é a diferença entre front-end e back-end no desenvolvimento de aplicações? **AWS Amazon**. Disponivel em: <https://aws.amazon.com/pt/compare/the-difference-between-frontend-and-backend/>. Acesso em: 13 setembro 2024.

BREGOLA, C. Figma: o que é a ferramenta, design e como usar. **Alura**, 2024. Disponivel em: <https://www.alura.com.br/artigos/figma?srsltid=AfmBOooKc3dBBUQa9R2rO\_QmRL-cMRkpt0bZnRGx6jVJk34xSoGq4KkU>. Acesso em: 19 Setembro 2024.

CARVALHO, C. O que é Python? — um guia completo para iniciar nessa linguagem de programação. **Alura**, 2024. Disponivel em: <https://www.alura.com.br/artigos/python>. Acesso em: 19 Setembro 2024.

CELCOIN. Regras de negócio: o que são, importância e como utilizar. **Celcoin**, 2023. Disponivel em: <https://www.celcoin.com.br/news/regras-de-negocio/?bot\_cel\_cash=1>. Acesso em: 12 Setembro 2024.

COLABORADOR, T. MySQL . **ComputerWeekly**. Disponivel em: <https://www.computerweekly.com/br/definicoe/MySQL>. Acesso em: 19 Setembro 2024.

CUNHA, F. Requisitos funcionais e não funcionais: o que são? **Mestres da Web**. Disponivel em: <https://www.mestresdaweb.com.br/tecnologias/requisitos-funcionais-e-nao-funcionais-o-que-sao>. Acesso em: 12 Setembro 2024.

EBAC. O que é JavaScript e como funciona? **EBAC**, 2023. Disponivel em: <https://ebaconline.com.br/blog/o-que-e-javascript-seo>. Acesso em: 13 Setembro 2024.

EQUIPE PM3. Banco de dados: o que é, importância, tipos e como criar um. **PM3**, 2023. Disponivel em: <https://www.cursospm3.com.br/blog/banco-de-dados/>. Acesso em: 19 Setembro 2024.

EQUIPE TOTVS. Front end: O que é, como funciona e qual a importância. **TOTVS**, 2024. Disponivel em: <https://www.totvs.com/blog/developers/front-end/>. Acesso em: 19 Setembro 2024.

EWALLY. Back-end: O Que É, Para Que Serve e Quais Suas Linguagens? **Ewally**. Disponivel em: <https://www.ewally.com.br/blog/ajudando-sua-empresa/backend>. Acesso em: 13 Setembro 2024.

FABRO, C. O que é API e para que serve? **TechTudo**, 2020. Disponivel em: <https://www.techtudo.com.br/listas/2020/06/o-que-e-api-e-para-que-serve-cinco-perguntas-e-respostas.ghtml>. Acesso em: 19 Setembro 2024.

FERREIRA, K. Figma: o que é e como usar na criação de interfaces de produtos. **Tera Blog**. Disponivel em: <https://blog.somostera.com/ux-design/figma>. Acesso em: 19 Setembro 2024.

HASHTAG. O que é Flask no Python: Micro-Framework de Desenvolvimento Web no Python. **Hashtag**, 2024. Disponivel em: <https://www.hashtagtreinamentos.com/o-que-e-flask-python>. Acesso em: 19 Setembro 2024.

INVENT+. Invent+. **Figma**, 2024. Disponivel em: <https://www.figma.com/design/hf9l2FZUspGQDKvJ1Evpri/Almoxarifado-v.02?node-id=0-1&node-type=canvas>.

L., A. O Que é AJAX e Como Funciona? **Hostinger Tutoriais**, 2023. Disponivel em: <https://www.hostinger.com.br/tutoriais/o-que-e-ajax>. Acesso em: 13 Setembro 2024.

L., A. O Que é HTML: O Guia Definitivo para Iniciantes. **Hostinger Tutorials**, 2023. Disponivel em: <https://www.hostinger.com.br/tutoriais/o-que-e-html-conceitos-basicos>. Acesso em: 13 Setembro 2024.

MARQUES, R. O que é Bootstrap: Tudo sobre este Framework. **Home Host**, 2023. Disponivel em: <https://www.homehost.com.br/blog/tutoriais/o-que-e-bootstrap/>. Acesso em: 13 Setembro 2024.

MARQUES, R. O que é um Levantamento de Requisitos de Software? **Cedro Technologies**, 2023. Disponivel em: <https://www.cedrotech.com/blog/levantamento-de-requisitos-e-desenvolvimento-de-softwares/>. Acesso em: 13 Setembro 2024.

MDN WEBS DOCS. O que é JavaScript? **Mdn Webs Docs**. Disponivel em: <https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/First\_steps/What\_is\_JavaScript>. Acesso em: 13 Setembro 2024.

MIRANDA, L. F. Requisitos funcionais e não funcionais: o que são, diferenças e exemplos. **Quero Bolsa**. Disponivel em: <https://querobolsa.com.br/revista/requisitos-funcionais-e-nao-funcionais>. Acesso em: 12 Setembro 2024.

NORMAS ABNT. Glossário TCC: como fazer na ABNT? **Normas ABNT**. Disponivel em: <https://www.normasabnt.org/glossario-tcc/>. Acesso em: 13 Setembro 2024.

O que é CSS? Guia Básico para Iniciantes. **Hostinger Tutorials**, 2022. Disponivel em: <https://www.hostinger.com.br/tutoriais/o-que-e-css-guia-basico-de-css>. Acesso em: 13 Setembro 2024.

PYTHON. Python Geral. **Python**. Disponivel em: <https://docs.python.org/pt-br/dev/faq/general.html>. Acesso em: 19 Setembro 2024.

ROGÉRIO. O que é o AJAX? **Dev Media**. Disponivel em: <https://www.devmedia.com.br/o-que-e-o-ajax/6702>. Acesso em: 13 Setembro 2024.

S, D. Flask Python: como usar e qual sua função? **Home Host**, 2024. Disponivel em: <https://www.homehost.com.br/blog/pythondjango/flask-python/>. Acesso em: 19 Setembro 2024.

SANTANA, B. O Que é Python: Conheça Uma das Linguagens de Programação mais Populares do Mundo. **Hostinger**, 2024. Disponivel em: <https://www.hostinger.com.br/tutoriais/python-o-que-e>. Acesso em: 06 Setembro 2024.

SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL. **Almoxarife**. São Paulo: SENAI-SP Editora, 2014.

SOUTO, O. O que é Front-end Back-end e Full Stack - aprenda as diferenças entre essas áreas. **Alura**, 2024. Disponivel em: <https://www.alura.com.br/artigos/o-que-e-front-end-e-back-end?srsltid=AfmBOoqwnoHO7cfKUcJiV0shTurrQmnxOsFzLly8VH9GPtxKb7a7iTX4>. Acesso em: 13 Setembro 2024.

SOUTO, O. O que é Front-end Back-end e Full Stack - aprenda as diferenças entre essas áreas. **Alura**, 2024. Disponivel em: <https://www.alura.com.br/artigos/o-que-e-front-end-e-back-end>. Acesso em: 19 Setembro 2024.

SOUZA, I. D. Banco de dados: saiba o que é, os tipos e a importância para o site da sua empresa. **Rock Content**, 2023. Disponivel em: <https://rockcontent.com/br/blog/banco-de-dados/>. Acesso em: 19 Setembro 2024.

THE PYTHON SOFTWARE FOUNDATION. Python Geral. **Python Software Foundation**. Disponivel em: <https://docs.python.org/pt-br/dev/faq/general.html#what-is-python>. Acesso em: 06 Setembro 2024.

**GLOSSÁRIO**

**Framework**

**Backend**

**Frontend**

Bcrypt

**APÊNDICE A – Cadastro de materiais**

Esse código em Python é parte de uma aplicação web que permite ao usuário cadastrar materiais em um sistema. Ele verifica se o usuário tem permissão, coleta os dados do formulário, tenta salvar esses dados em um banco de dados e lida com erros e confirmações de maneira adequada.

@app.route('/cadastro\_material', methods=['GET', 'POST'])

def cadastro\_material():

    if 'user\_cargo' not in session or session['user\_cargo'] != 'almoxarifado':

        return redirect(url\_for('solicitante'))

    if request.method == 'POST':

        descricao = request.form['descricao']

        categoria = request.form['categoria']

        estoque\_minimo = int(request.form['estoque\_minimo'] or 0)

        estoque\_maximo = int(request.form['estoque\_maximo'] or 0)

        conexao = conectar\_banco\_dados()

        cursor = conexao.cursor()

        try:

            query\_insert = ("INSERT INTO materials (descricao, categoria, estoque\_minimo, estoque\_maximo) "

                            "VALUES (%s, %s, %s, %s)")

            cursor.execute(query\_insert, (descricao, categoria, estoque\_minimo, estoque\_maximo))

            conexao.commit()

            flash('Material cadastrado com sucesso!', 'success')

            return redirect(url\_for('cadastro\_material'))

        except:

            return redirect(url\_for('cadastro\_material'))

        finally:

            cursor.close()

            conexao.close()

    return render\_template('funcoes/cadastro\_material.html')

Esse código HTML cria uma página de cadastro de materiais em uma aplicação web. Ele inclui um formulário onde os usuários podem inserir informações sobre um novo material, como o nome, a categoria e outros detalhes. É uma parte importante de um sistema onde os usuários podem adicionar novos itens a um inventário.

<!DOCTYPE html>

<html lang="pt-BR">

<head>

    <meta charset="UTF-8">

    <link rel="stylesheet" href="{{ url\_for('static', filename='styles/cadastro-materias.css') }}">

    <link rel="stylesheet" href="{{ url\_for('static', filename='styles/variaveis.css') }}">

    <link rel="stylesheet" href="{{ url\_for('static', filename='styles/medias/media-cadastro-item.css') }}">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Cadastro de Material</title>

</head>

<body>

    {% include '/dashboard/dashboard.html' %}

    <main>

        <section class="cadastro">

            <div class="campo-cadastro">

                <form class="form-cadastro" action="{{ url\_for('cadastro\_material') }}" method="POST">

                    <div class="titulo-cadastro">

                        <img src="/static/img/logo01.png" alt="logo">

                        <h2>Cadastro de Material</h2>

                    </div>

                    <div class="input\_nome">

                        <input type="text" id="descricao" name="descricao" placeholder="Nome do produto" required>

                    </div>

                    <div class="input\_categoria">

                        <select id="categoria" name="categoria" required>

                            <option value="">Categoria</option>

                            <option value="consumiveis">Consumíveis</option>

                            <option value="ferramentas">Ferramentas</option>

                            <option value="equipamentos">Equipamentos</option>

                        </select>

                    </div>

                    <div class="input\_estoque">

                        <input type="number" id="estoque\_minimo" name="estoque\_minimo" min="0" placeholder="Estoque Mínimo" required>

                        <input type="number" id="estoque\_maximo" name="estoque\_maximo" min="0" placeholder="Estoque Máximo" required>

                    </div>

                    <div class="container-btn">

                        <button class="btn-cadastro" type="submit">Cadastrar</button>

                    </div>

                    {% with messages = get\_flashed\_messages() %}

                        {% if messages %}

                            <ul class="flashes">

                            {% for message in messages %}

                                <li>{{ message }}</li>

                            {% endfor %}

                            </ul>

                        {% endif %}

                    {% endwith %}

                </form>

            </div>

        </section>

    </main>

</body>

</html>

Esse código CSS define como a página de cadastro de materiais deve ser exibida. Além disso, inclui ajustes para garantir que a página fique responsiva em diferentes dispositivos, como celulares e tablets.

/\*

   Normalização do Navegador

   - Remove margens e paddings padrão de todos os elementos.

   - Define a caixa de todos os elementos como "box-sizing: border-box".

   - Remove o estilo de lista e a sublinha de links.

   - Define a fonte padrão 'Amatic SC'.

\*/

\* {

    margin: 0;

    padding: 0;

    box-sizing: border-box;

    list-style: none;

    text-decoration: none;

    border: none;

}

/\*

   Normalização do HTML e body

   - Define a largura e altura como 100% da viewport.

   - Ajusta a unidade rem para que 1rem seja equivalente a 10px.

   - Remove o overflow horizontal.

\*/

/\*

   Seção de Cadastro

   - Centraliza o conteúdo da seção usando flexbox.

   - Define a largura e altura para ocupar toda a tela.

   - Aplica um fundo cinza claro.

\*/

.cadastro {

    display: flex;

    justify-content: center;

    align-items: center;

    width: 100%;

    height: 100vh;

    background: rgb(0,71,102);

    background: linear-gradient(6deg, rgba(0,71,102,1) 12%, rgba(177,196,204,1) 100%);

}

/\*

   Campo de Cadastro

   - Centraliza o conteúdo do campo de cadastro.

   - Aplica um fundo verde com padding.

   - Define uma largura máxima de 900px para telas grandes e 100% para telas pequenas.

\*/

/\* Campo de Cadastro \*/

.campo-cadastro {

    display: flex;

    flex-direction: column;

    justify-content: center;

    align-items: center;

    background-color: var(--branco);

    padding: 50px;

    border-radius: 8px;

    max-width: 900px;

    width: 100%;

    /\* Adicionando box-shadow para criar um efeito de profundidade \*/

    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2); /\* Sombra leve \*/

    transition: box-shadow 0.3s ease; /\* Transição suave para efeitos de foco \*/

}

.campo-cadastro:hover {

    /\* Sombra mais intensa ao passar o mouse \*/

    box-shadow: 0 6px 12px rgba(0, 0, 0, 0.3);

}

/\*

   Formulário de Cadastro

   - Ajusta o formulário para ocupar toda a largura disponível.

   - Alinha o texto à esquerda.

\*/

.form-cadastro {

    text-align: left;

    width: 100%;

}

/\*

   Título do Cadastro com Logo

   - Centraliza verticalmente a logo e o título.

   - Aplica espaçamento entre a logo e o texto.

\*/

.titulo-cadastro {

    display: flex;

    align-items: center;

    justify-content: center;

    margin-bottom: 30px;

}

.titulo-cadastro img {

    width: 90px;

    height: 90px;

    margin-right: 10px;

}

.titulo-cadastro h2 {

    font-size: 24px;

    margin-bottom: 0;

}

/\*

   Estilos para Inputs e Select

   - Alinha os campos em coluna.

   - Define padding e largura total para inputs e selects.

   - Aplica bordas arredondadas e ajusta a tipografia.

\*/

.input\_nome,

.input-descricao,

.input\_categoria {

    display: flex;

    flex-direction: column;

    margin-bottom: 20px;

    width: 100%;

}

input, select{

    background-color: rgb(243, 243, 243);

    color: black;

}

.input\_nome input,

.input-descricao input,

.input\_categoria select {

    width: 100%;

    padding: 15px;

    border-radius: 4px;

    border: 1px solid var(--cinza-azulado);

    font-size: 16px;

}

/\*

   Inputs de Estoque

   - Define os inputs lado a lado com um espaçamento entre eles.

   - Ajusta o tamanho e o estilo dos inputs de estoque mínimo e máximo.

\*/

.input\_estoque {

    display: flex;

    justify-content: space-between;

    margin-bottom: 30px;

    gap: 20px;

}

.input\_estoque input {

    width: 48%;

    padding: 15px;

    border-radius: 4px;

    border: 1px solid var(--cinza-azulado);

    font-size: 16px;

}

/\*

   Estilos do Botão de Cadastro

   - Centraliza o botão de cadastro.

   - Aplica estilo de cursor, cores e um efeito de hover.

   - Define a transição suave para o efeito de hover.

\*/

.btn-cadastro {

    justify-content: center;

    align-items: flex-end;

    width: 25%;

    padding: 15px;

    font-size: 16px;

    cursor: pointer;

    background-color: var(--azul-escuro);

    color: var(--branco);

    border-radius: 5px;

    transition: all 0.3s ease;

}

.btn-cadastro:hover {

    background-color: var(--azul-claro);

    transform: scale(1.05);

}

.container-btn{

    display: flex;

    width: 100%;

    justify-content: flex-end;

}

.flash-messages {

    position: fixed;

    top: 35px;

    left: 50%;

    transform: translate(-50%, -50%);

    z-index: 1000;

    width: 90%; /\* Ajuste a largura conforme necessário \*/

    max-width: 400px; /\* Limite a largura máxima \*/

    text-align: center; /\* Centralizar o texto \*/

    font-size: 18px;

}

.flash {

    padding: 15px;

    margin: 10px 0;

    border-radius: 5px;

    color: #fff;

    box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1); /\* Sombra opcional \*/

}

.flash.error {

    background-color: #e74c3c; /\* Vermelho \*/

}

.flash.success {

    background-color: #2ecc71; /\* Verde \*/

}

/\* Estilos para telas grandes (PCs e notebooks) \*/

@media (min-width: 1024px) {

    .primeiro-login {

        flex-direction: row;

        padding: 40px;

    }

    .banner-primeirologin {

        flex: 1;

        max-width: 600px;

        background: url('/static/img/fotoprimeirologin.jpg') no-repeat center center;

        background-size: cover;

        margin-right: 40px;

    }

    .primeiro-login02 {

        max-width: 500px;

    }

}

/\* Media Queries para Tornar a Página Responsiva \*/

/\* Telas Médias (Tablets) \*/

@media (max-width: 768px) {

    .cadastro {

        background-color: var(--branco); /\* Mantém o fundo verde \*/

        height: auto; /\* Permite o conteúdo expandir verticalmente \*/

    }

    .campo-cadastro {

        background-color: var(--branco); /\* Mantém o fundo verde no campo de cadastro \*/

        padding: 20px; /\* Ajusta o padding para telas menores \*/

        border-radius: 0; /\* remove bordas arredondadas \*/

        max-width: 100%; /\* Ocupa toda a tela \*/

        height: 100vh; /\* O campo de cadastro ocupará toda a altura da tela \*/

    }

    .input\_estoque {

        flex-direction: column; /\* Inputs empilhados verticalmente \*/

    }

    .input\_estoque input {

        width: 100%; /\* Inputs ocupam 100% da largura no modo coluna \*/

    }

    .btn-cadastro {

        font-size: 1.2rem;

        padding: 12px;

    }

}

/\* Telas Pequenas (Celulares) \*/

@media (max-width: 480px) {

    .cadastro {

        background-color: var(--branco); /\* Mantém o fundo verde \*/

        height: auto; /\* Permite o conteúdo expandir verticalmente \*/

    }

    .campo-cadastro {

        background-color: var(--branco); /\* Mantém o fundo verde no campo de cadastro \*/

        padding: 10px; /\* Ajusta o padding para telas menores \*/

        border-radius: 0; /\* Remove bordas arredondadas \*/

        max-width: 100%; /\* Ocupa toda a tela \*/

        height: 90vh;

    }

    .titulo-cadastro h2 {

        font-size: 2rem;

    }

    .input\_nome input,

    .input-descricao input,

    .input\_categoria select {

        padding: 12px;

        font-size: 1.2rem;

    }

    .input\_estoque input {

        padding: 12px;

        font-size: 1.2rem;

    }

    .btn-cadastro {

        font-size: 1.2rem;

        padding: 10px;

        margin-right: 12px;

    }

}

**ANEXO A – Título do anexo**