

FATEC RIO PRETO – Faculdade de Tecnologia de São José do Rio Preto

Liam Martingo Rodrigues

Victor Pasiani Marin

Arquitetura e Sistema de Gerenciamento e Análise de Estoque para visualização  
e adição de novos produtos para micro e pequenas farmacêuticas

Trabalho de Graduação

Prof. Dr. Carlos Magnus Carlson Filho

Orientador

São José do Rio Preto, SP – Brasil

2024

## RESUMO

O objetivo deste trabalho é desenvolver um software de controle de estoque utilizando SQL, Python e JavaScript. Este sistema visa melhorar a gestão e a análise de estoque em farmácias, garantindo um gerenciamento de recursos mais eficiente. Espera-se que a solução proposta ofereça benefícios significativos, como a redução de desperdícios, otimização do tempo de reposição de produtos e aumento da precisão na gestão dos inventários.

Palavras-chave: Controle de estoque, Farmácia, Análise de estoque, SQL, Python

## ABSTRACT

The objective of this study has been to develop inventory control software using SQL, Python and JavaScript. This system has aimed to enhance inventory management and analysis in pharmacies, ensuring more efficient resource management. The proposed solution has been expected to provide significant benefits, such as waste reduction, optimization of product replenishment times, and increased accuracy in inventory management.

Keywords: Inventory control, Pharmacy, Inventory analysis, SQL, Python

### 1. Introdução

A gestão eficiente de estoque é um dos pilares para o sucesso de qualquer negócio, dado o impacto direto que tem na satisfação dos clientes e na lucratividade do negócio. A manutenção de um estoque adequado é essencial para evitar perdas devido à expiração de produtos ou ao desabastecimento, situações que podem comprometer a confiança dos consumidores e resultar em prejuízos financeiros.

No entanto, muitas farmácias ainda enfrentam dificuldades em manter um controle rigoroso dos seus inventários, seja pela falta de ferramentas tecnológicas adequadas ou pela complexidade das operações envolvidas. (Garcia, Pereira & Osório, 2009).

Esse cenário destaca a necessidade de soluções tecnológicas que possam facilitar a gestão de estoques, integrando análise de dados e controle em tempo real.

Este trabalho propõe o desenvolvimento de um software de controle de estoque específico para farmácias, com o objetivo de otimizar a gestão dos recursos e melhorar a precisão na análise dos inventários. A solução visa não apenas facilitar o processo de reposição de produtos, deste modo auxiliando na tomada de decisão.

## 2. Metodologia

O desenvolvimento do software iniciará-se com o levantamento de requisitos, será realizado por meio de pesquisas teóricas e análise de sistemas de controle de estoque existentes no mercado. A partir dessas informações, serão definidos os requisitos funcionais e não funcionais, que orientarão o design do sistema. O backend será desenvolvido em Python, com a ajuda do framework Django, enquanto o frontend será implementado em JavaScript. O banco de dados será estruturado em SQL Server, assegurando a integridade e a eficiência no gerenciamento dos dados.

Toda a implementação ocorrerá em ambiente acadêmico, com a criação de protótipos que serão submetidos a testes internos, incluindo testes unitários e de integração, para verificar a funcionalidade e a usabilidade do sistema. Embora o software não seja aplicado em um ambiente real de farmácia, sua eficácia será avaliada com base em simulações e estudos de caso hipotéticos, visando demonstrar sua capacidade de melhorar o controle de estoque e a gestão de recursos em farmácias.

## 3. Fundamentação Teórica

### 3.1 Controle de Estoque

O controle de estoques requer um planejamento estratégico cuidadoso, em que a manutenção intencional de estoques pode trazer benefícios, enquanto a ausência ou falhas no planejamento revelam inadequações que devem ser corrigidas prontamente, com o auxílio de ferramentas modernas de previsão e logística (ROSA, 2015).

A previsão de demanda e a reposição eficiente de produtos são fundamentais para manter o fluxo adequado de mercadorias em qualquer operação, evitando faltas ou excessos de estoque e garantindo a continuidade das atividades empresariais.

### 3.2 Farmacos

Segundo Garcia, Pereira e Osório (2009), todo desafio de um gestor de estocagem é a consistência e organização no estoque, e gerir no segmento farmacêutico, medicamentos e cosméticos não seria diferente. Trabalhar com esse tipo de logística precisa se tornar preciso e organizado, pois necessita de receituário prescrito e outras demandas específicas.

Logo esse segmento usa métodos e esquemas precisos para melhor eficiência e organização num ambiente farmacêutico, por conseguinte, um sistema e uma arquitetura de uma aplicação precisam estar alinhados com esses princípios, tornando prático e com qualidade.

### 4. Trabalhos Similares

O trabalho de Costa et al. (2018) foca no desenvolvimento de um sistema de gerenciamento de estoque para uma empresa que enfrentava graves falhas em seu controle de produtos. Com base em uma pesquisa qualitativa e exploratória, os autores identificaram problemas como a ausência de suporte tecnológico e a inconsistência de dados. A solução proposta é um software específico para melhorar o controle de produtos, com o objetivo de otimizar os processos de estoque, reduzir custos operacionais e gerar retorno financeiro para a empresa.

O estudo de Silva et al. (2015) aborda o desenvolvimento de um software voltado para farmácias, com foco em rapidez e facilidade de programação. O trabalho foi motivado pela constatação de que os softwares existentes apresentam falhas e não atendem plenamente às necessidades desses estabelecimentos. O objetivo é criar um sistema que melhore o atendimento ao usuário, otimize o controle de estoque e corrija bugs que afetam a eficiência. O software oferece ferramentas rápidas e simples, com uma interface para consulta de informações detalhadas sobre medicamentos, além de melhorar a comunicação entre farmacêuticos, clientes e fornecedores, tornando o processo mais ágil.

O trabalho de Oliveira (2008) aborda o desenvolvimento de um sistema de gerenciamento para drogarias, motivado pela crescente demanda por controle informatizado devido ao aumento no número de clientes e na variedade de produtos. O sistema tem como objetivo otimizar o controle de estoque de medicamentos, débitos e dados cadastrais, além de gerar relatórios que facilitam o atendimento, tornando o ambiente de trabalho mais eficiente e seguro. O projeto utiliza a metodologia de análise estruturada para a documentação, e o desenvolvimento ocorre no ambiente Delphi 7, com o banco de dados MySQL. Com a implementação do sistema, espera-se reduzir erros e melhorar o controle de informações na drogaria.

## 5. Desenvolvimento

### 5.1 Visão Geral

O sistema de controle de estoque desenvolvido tem como objetivo centralizar e automatizar a gestão de produtos em farmácias, facilitando o monitoramento das entradas, saídas e disponibilidade dos itens. A solução proposta permitirá uma visão em tempo real do estoque, ajudando a evitar rupturas de produtos e excesso de inventário, o que melhora a eficiência operacional e reduz custos.

#### 5.1.1 Controle de Estoque

O módulo de controle de estoque será o núcleo do sistema, responsável por gerenciar o fluxo de mercadorias, desde a entrada de novos itens até o monitoramento de saídas. Através de um banco de dados SQL, todas as informações sobre os produtos serão armazenadas de forma segura e acessível, possibilitando consultas detalhadas e relatórios personalizados. O sistema permitirá o registro de novas entradas de produtos, controle de lotes e validade, além da emissão de alertas automáticos para produtos com baixo estoque ou próximos ao vencimento.

A lógica de controle será implementada em Python, utilizando algoritmos que processam os dados do estoque, garantindo a precisão das informações e a atualização constante dos níveis de inventário. O sistema também permitirá que os usuários filtrem e organizem os dados conforme a necessidade, facilitando a tomada de decisões e otimizando os processos de reposição.

#### 5.1.2 Interface do Usuário

A interface será projetada em JavaScript, proporcionando uma experiência amigável e interativa para os usuários. O design da interface será intuitivo, facilitando o uso mesmo por profissionais com pouca familiaridade com tecnologia. A navegação será simples e permitirá o acesso rápido às principais funcionalidades do sistema, como cadastro de produtos, visualização do estoque atual, e emissão de relatórios.

#### 5.1.3 Relatórios e Análises

O sistema incluirá a funcionalidade de geração de relatórios periódicos sobre o desempenho do estoque. Esses relatórios podem ser configurados para exibir informações detalhadas sobre a movimentação de produtos, frequência de reposição, e itens com maior e menor demanda. Além disso, a integração entre Python e SQL permitirá que os relatórios sejam personalizados

de acordo com os critérios do usuário, oferecendo dados precisos para a tomada de decisões estratégicas.

## 5.2 Ferramentas e Tecnologias

No desenvolvimento do sistema foi utilizado um conjunto de ferramentas tecnológicas modernas, selecionadas pela sua robustez, eficiência e adequação às necessidades específicas do projeto. A combinação de SQL para a base de dados, Python para o back-end, e JavaScript para o front-end garante que o sistema seja eficiente, escalável e fácil de usar.

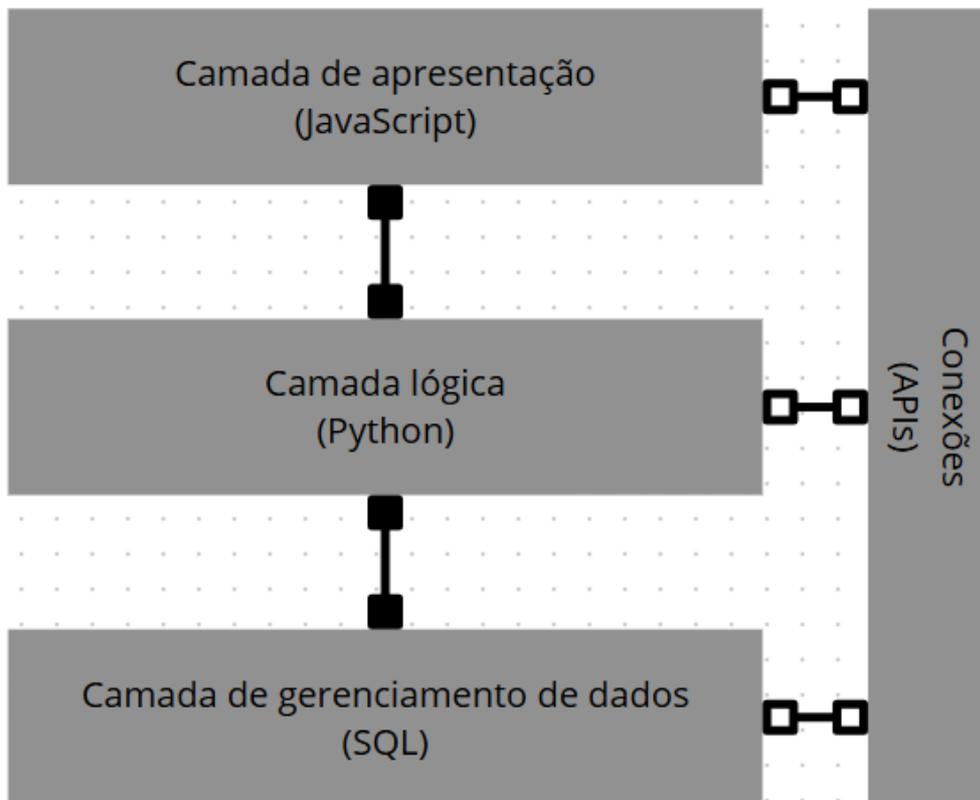
O SQL foi escolhido como a tecnologia de banco de dados por sua capacidade de armazenar, organizar e recuperar dados de forma confiável. Ele será usado para gerenciar as informações relativas ao estoque, incluindo o cadastro de produtos, movimentações de entrada e saída, e dados históricos. Além disso, o SQL oferece a segurança necessária para garantir a integridade dos dados, utilizando recursos como controle de transações, relacionamentos entre tabelas e restrições que evitam a redundância e erros nos registros.

Para o desenvolvimento da lógica de negócios do sistema, optou-se por Python devido à sua simplicidade e flexibilidade. Python oferece uma vasta gama de bibliotecas que facilitam o processamento de dados, integração com o banco de dados SQL e automação de processos como a geração de relatórios. A linguagem também facilita o desenvolvimento rápido de funções complexas, mantendo o código claro e fácil de manter, além de garantir a escalabilidade para futuras atualizações do sistema.

O front-end será desenvolvido utilizando JavaScript, uma linguagem amplamente utilizada para o desenvolvimento web que permite criar interfaces interativas e dinâmicas. JavaScript foi escolhido por sua capacidade de rodar diretamente no navegador, proporcionando uma experiência rápida e fluida ao usuário final. Com ele, será possível criar uma interface amigável para os funcionários da farmácia, permitindo o acesso rápido às funcionalidades do sistema, como cadastro de produtos, visualização do estoque e geração de relatórios.

## 5.3 Arquitetura

A arquitetura do sistema foi planejada para garantir escalabilidade e eficiência no processamento e armazenamento de dados. A solução é baseada em uma arquitetura multicamadas, que separa a apresentação, a lógica de negócios e o gerenciamento de dados. Esse modelo facilita a manutenção e futuras expansões do sistema, garantindo que as funcionalidades possam ser ajustadas sem comprometer o desempenho ou a integridade dos dados.



Fonte: Adaptado de Macorrati.net

### 5.3.1 Camada de Apresentação (Front-End)

A camada de apresentação será desenvolvida em JavaScript, responsável por gerar a interface gráfica que os usuários irão interagir. O objetivo principal desta camada é fornecer uma experiência de uso intuitiva e responsiva. A interface será construída de maneira a funcionar em dispositivos desktop e móveis, utilizando bibliotecas modernas de JavaScript, como React, para criar componentes dinâmicos e interativos.

Através dessa interface, os usuários poderão realizar diversas operações, como:

- Cadastro e consulta de produtos.
- Monitoramento de estoque em tempo real.
- Geração de relatórios personalizados.
- Configuração de alertas para níveis críticos de estoque ou produtos próximos ao vencimento.

A comunicação com a camada de lógica de negócios será realizada via APIs RESTful, enviando e recebendo dados em formato JSON, o que garante a flexibilidade e simplicidade nas interações entre cliente e servidor.

### 5.3.2 Camada de Lógica de Negócios (Back-End)

A lógica de negócios do sistema será implementada em Python, utilizando frameworks como Django para gerenciar as operações do sistema. Esta camada é responsável por processar todas as regras de negócios envolvidas no controle de estoque, como:

- Cálculo automático de quantidades de produtos em estoque com base em entradas e saídas.
- Verificação de lotes e datas de vencimento.
- Emissão de notificações automáticas para reabastecimento ou vencimento de produtos.
- Geração de relatórios detalhados baseados nas movimentações do estoque.

Além disso, a lógica de negócios também será responsável pela validação e integridade dos dados antes de serem armazenados no banco de dados, garantindo que as operações sejam executadas de maneira correta e segura.

### 5.3.3 Camada de gerenciamento de dados (Banco de Dados)

O banco de dados relacional será implementado em SQL e organizado para armazenar e gerenciar as informações essenciais do sistema de controle de estoque. A estrutura de dados será desenhada para garantir consistência, integridade e alta disponibilidade, além de facilitar o acesso rápido a informações críticas, como movimentações de estoque, validade de produtos, etc.

A estrutura do banco de dados será composta por tabelas que armazenam informações sobre:

- Produtos: detalhes como nome, lote, validade, preço, quantidade, etc.
- Movimentações de estoque: registros de entradas e saídas de produtos.
- Relatórios e logs: para auditoria e análise de desempenho.

### 5.3.4 Integração entre as Camadas

A comunicação entre a camada de apresentação (front-end), lógica de negócios (back-end) e banco de dados será feita por meio de APIs RESTful. Essas APIs serão responsáveis por permitir que o front-end envie solicitações ao back-end para executar operações como cadastro, consulta e atualização de produtos, e também para a geração de relatórios. O back-end, por sua vez, processará as solicitações e interagirá com o banco de dados para retornar as informações solicitadas.

Essa arquitetura em camadas oferece flexibilidade para modificar ou atualizar qualquer uma das partes do sistema sem afetar as demais.

### 5.3.5 Segurança e Autenticação

A segurança será um aspecto necessário na arquitetura do sistema. A seguinte medida será adotada:

- Autenticação de usuários: Utilizando o padrão de autenticação dado pelo backend para garantir que apenas usuários autorizados tenham acesso às funcionalidades críticas do sistema. Logo, utilizando o padrão de autenticação fornecido pela tecnologia integrada do framework Django e Django Rest Framework, é possível utilizar uma estrutura de administradores de autenticação para os usuários trabalhando em níveis de usuários e escala, deixando assim bastante robusto e seguro.

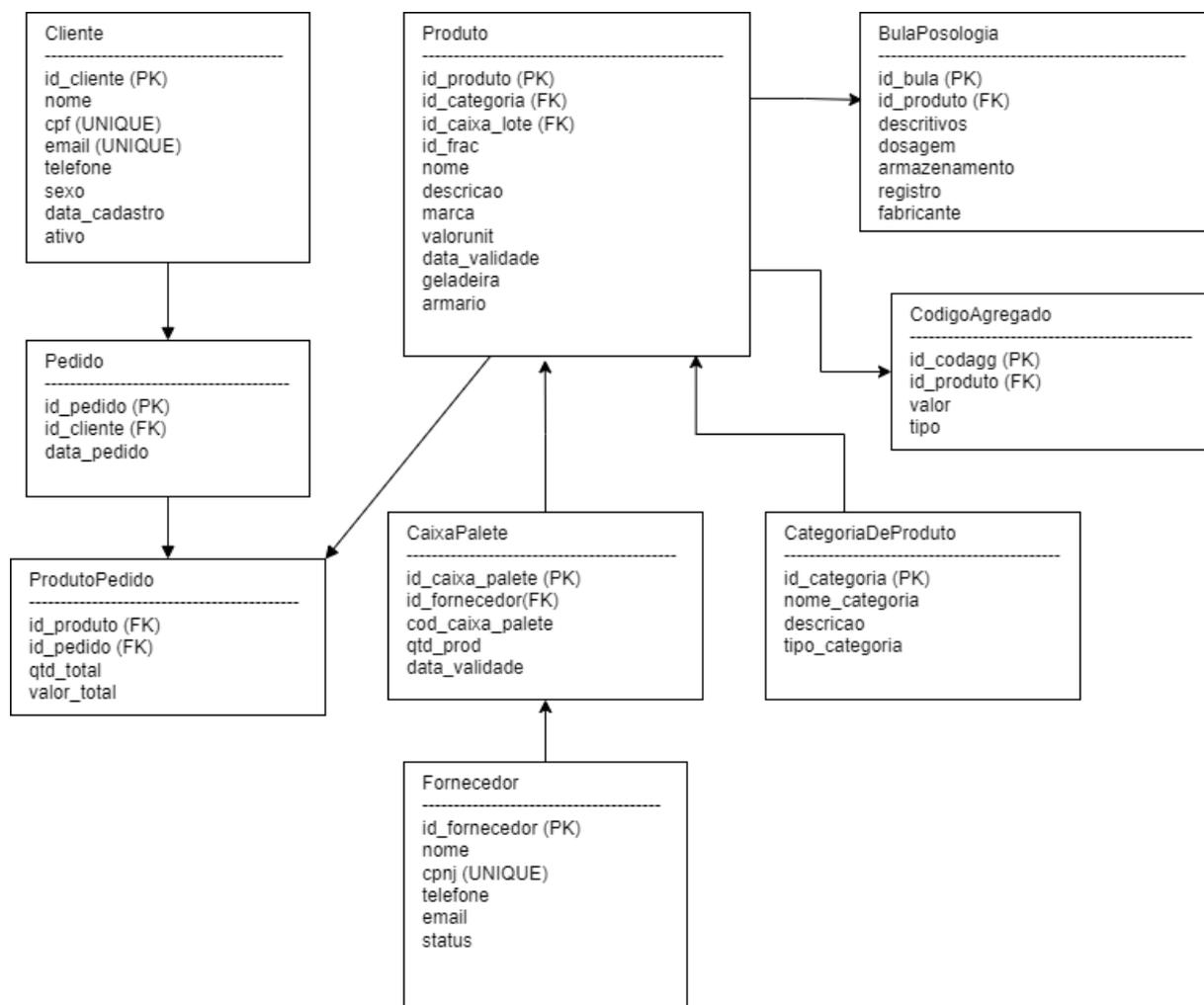
Essa arquitetura modular e distribuída garante que o sistema seja eficiente, fácil de manter e expandir, e capaz de lidar com as demandas do controle de estoque de farmácias, oferecendo confiabilidade e desempenho.

## 5.4 Modelagem

### 5.4.1 Diagrama Entidade-Relacionamento (DER)

Trabalhando com armazenagem e estoque sua estrutura de gerenciamento de dados precisa ser bem organizada e operando com cada processo dentro do ambiente farmacêutico, logo, foi pensando apenas para estocagem, nisso, vendas e compras não estão totalmente mapeados no diagrama, portanto, trabalhar com o medicamento, sua quantidade mantida, se foi removido e registro de compras são exemplificados.

Adicionalmente, os itens relacionados com a tabela Cliente tratam-se de saída de produtos, enquanto os itens relacionados com a tabela Fornecedores referem-se a entradas de produtos, permitindo uma organização clara entre os fluxos de entrada e saída no estoque. A imagem abaixo estabelece as entidades e seus relacionamentos formados:



Fonte: Elaborada pelos autores.

### 5.4.2 Telas

O desenvolvimento das telas foi estruturado em camadas no padrão de aplicações como todo sistema desse porte, ele possui interface para autenticação como login e cadastro de usuário,

além de uma tela para configuração de usuário, nos próximos tópicos são descritos as telas principais do sistema e suas peculiaridades.

#### 5.4.2.1 Tela de Dashboard

A tela principal é o Dashboard, que fornece uma visão geral das movimentações de estoque. Nela, os usuários encontram:

- **Indicadores Resumidos:** Mostram o total de itens, a quantidade de entradas e saídas recentes, permitindo uma visualização rápida dos dados essenciais.
- **Gráfico de Volume de Estoque:** Exibe a movimentação de estoque ao longo do tempo, categorizando as entradas e saídas de produtos para facilitar a análise visual e a tomada de decisões.

Esses elementos otimizam o acompanhamento e o controle do estoque, oferecendo informações visuais de forma intuitiva.

#### 5.4.2.2 Tela de Inventário

Nesta tela, o usuário tem acesso a uma lista detalhada de todos os produtos em estoque, incluindo informações como descrição, categoria, quantidade disponível, preço, data de validade e fornecedor. Esta tela facilita a gestão e a localização de produtos específicos, bem como o acompanhamento de datas importantes para evitar desperdícios.

#### 5.4.2.3 Tela de Relatórios

Esta tela tem um papel fundamental, pois ela irá mostrar com detalhes toda a parte gráfica e de dados que o usuário precisa, permitindo que ele consiga ter métricas para avaliar melhor seu estoque e permitindo se adequar às demandas de novos produtos e reposições. Trabalhando com indicadores mais detalhados, facilita ao usuário ter uma melhor visão do que está acontecendo e mais importante, conseguir prevenir falhas e/ou propor melhorias no estoque.

Na tela em exemplo os seguintes indicadores base são:

- **Produtos com maior saída:** Este indicador é para mostrar quais produtos estão sendo comprados, ou seja, removidos do estoque, mostrando que ele é um produto requisitado e bastante necessário. Permite que se concentre mais nesse produtos para repor em tempo recorde do que outros que podem ter um custo maior, assim podendo viabilizar mais compras e diminuição de custos.
- **Quantidade de Estoque e Estoque de Segurança:** Este indicador é o mais simples e mais necessário, pois mostra a quantidade total de estoque de um produto, podendo saber se ele está acabando e necessário repor ou se é um produto com pouca demanda, sem possuir fluxo de entrada e saída muito alto. Portanto, saber dessas informações pode ajudar bastante na tomada de decisão para otimizar o fluxo de entrada e saída dos produtos.

- Fluxo de Estoque: Indicador de fluxo, é outro também bastante importante, pois ele determina como está o fluxo da farmácia, se há um fluxo de vendas razoável para semana ou mês, auxiliando a trabalhar com quais produtos podem ser vendidos em tal época.

Dado este contexto, esses indicadores são os determinados ‘padrões’, indicadores mais significativos, pois são o que o gestor mais precisa saber, porém também foi trabalhado permitir personalização nesta tela, sendo assim, você pode exportar essas informações, como também pode personalizar o relatório que precisa, permitindo especificações para os dados.

## 5.5 Protótipo

Neste tópico mostrará como está estruturado o protótipo de aplicação web desenvolvido para este artigo. Como comentado no tópico anterior, cada tela foi descrita e suas respectivas funções para eficiência no fluxo e organização do estoque.

Na figura abaixo é mostrado a tela de dashboard, primeira tela de visualização do sistema.



Fonte: Elaborada pelos autores.

Foi usado uma construção de interface mais intuitiva e padrão, pois vários outros projetos maiores de sistemas mais robustos possuem este estilo e estrutura, por isso a familiaridade pode ser mais atrativa para visualizar dados de maneira mais rápida e eficiente.

Na figura abaixo trata do inventário do estoque, onde a tela de interação com o usuário, permitindo colocar, remover e pesquisar itens armazenados.

Número	Nome	Descrição	Marca	Valor Unitário	Data de Validade	Quantidade
001	Produto A	Descrição do Produto A	Marca A	R\$ 10,00	01/01/2025	100
002	Produto B	Descrição do Produto B	Marca B	R\$ 15,00	15/03/2025	150
003	Produto C	Descrição do Produto C	Marca C	R\$ 20,00	30/06/2025	200
004	Produto D	Descrição do Produto D	Marca D	R\$ 25,00	10/09/2025	250
005	Produto E	Descrição do Produto E	Marca E	R\$ 30,00	20/12/2025	300
006	Produto F	Descrição do Produto F	Marca F	R\$ 35,00	05/02/2026	350
007	Produto G	Descrição do Produto G	Marca G	R\$ 40,00	18/04/2026	400
008	Produto H	Descrição do Produto H	Marca H	R\$ 45,00	27/07/2026	450
009	Produto I	Descrição do Produto I	Marca I	R\$ 50,00	12/10/2026	500
010	Produto J	Descrição do Produto J	Marca J	R\$ 55,00	31/12/2026	550

Fonte: Elaborado pelos autores.

Esta tela em específico possui a finalidade de mostrar todos produtos armazenados, mostrando as principais informações do produto.

Na figura abaixo mostra a tela que compõe a parte mais importante e objetivo principal da aplicação descrita neste artigo, trabalha mostrando os dados e principais métricas que o usuário precisa ao organizar o estoque de maneira coerente e precisa.



Fonte: Elaborada pelos autores.

A tela oferece uma interface intuitiva, com navegação simplificada pelos módulos na barra lateral e opções de personalização por meio de abas e checklists, permitindo escolher qual métrica precisa ser analisada unicamente. Os gráficos destacam dados de forma clara,

auxiliando na análise visual, enquanto o botão "Exportar" facilita o uso externo dos relatórios. O design responsivo e organizado aprimora a experiência do usuário e a eficiência na tomada de decisões.

## 5.6 Testes

### 5.6.1 Testes Unitários

Os testes unitários são fundamentais para descobrir barreiras em nível de código permitindo melhor aproveitamento do sistema e assim impedir erros recorrentes e simples. No projeto, foram realizados testes unitários em cada uma das rotas acessadas em primeiro e segundo plano, sendo dividido em testes no desenvolvimento de interface e desenvolvimento da regra de negócio.

#### 5.6.1.1 Testes unitários na camada de lógica de negócio

Os testes na regra de negócio implica em saber se todas as rotas e camadas de banco de dados e requisições estão bem estruturadas e sem micro erros, como exemplo, dados extraídos em estado incompleto quando é feita a chamada no método GET na rota de inventário.

Estrutura utilizado para validar o sistema:

1. Testes generalistas em Classes e Funções de código: é feito os testes unitários gerais, utilizando como base as classes e funções que constituem as rotas. Verificando se a lógica construída do retorno da informação ou não nas chamadas estão corretos e os resultados esperados batem.
2. Trabalhando com respostas das rotas: estes testes já possuem uma especificidade, visualizando se a verificação das respostas das rotas estão corretas. Por exemplo, ao realizar uma chamada no método GET se informa que está vindo a resposta 200, ao executar uma chamada correta.

#### 5.6.1.2 Testes unitários na camada de apresentação

Estrutura principal dos testes elaborados para interfaces do sistema:

- Renderização do Componente: Utiliza-se a função `render` da React Testing Library para montar o componente em um ambiente de teste virtual.
- Interação com o Componente: Simular eventos do usuário, como cliques ou entradas de texto, usando funções como `userEvent`.
- Asserções: Verifica-se se o componente responde conforme o esperado, usando métodos de asserção fornecidos pelo Jest, como `expect`.

### 5.6.2 Testes de integração

A verificação da comunicação entre frontend e backend consiste em garantir que as requisições originadas no frontend sejam corretamente recebidas, processadas e respondidas pelo backend, e que as respostas sejam devidamente interpretadas pelo frontend. Esse processo assegura o

pleno funcionamento das telas principais, validando operações críticas como a visualização de dados, adição de produtos e geração de métricas personalizadas nas interfaces de Dashboard, Inventário e Relatórios.

Além disso, envolve a identificação e correção de inconsistências para resolver problemas de compatibilidade entre os dados esperados pelo frontend e os retornados pelo backend, incluindo formatos de dados incompatíveis ou endpoints configurados incorretamente.

## 6. Conclusão

### 6.1 Resultados

O sistema de controle de estoque desenvolvido atendeu às expectativas iniciais, oferecendo uma plataforma eficiente e intuitiva para o gerenciamento de entradas e saídas de produtos. A inclusão de gráficos no dashboard permitiu uma análise visual dos dados, melhorando a tomada de decisões e reduzindo o risco de perdas.

### 6.2 Trabalhos Futuros

Como aprimoramentos futuros, possibilita-se a implementação de novos módulos para ampliar as funcionalidades, tornando o sistema semelhante a soluções como o SAP. A integração com dispositivos móveis permanece essencial para acessibilidade e controle remoto. Além disso, considera-se a possibilidade de implementar o sistema em um ambiente produtivo, avaliando seu desempenho e impacto no cenário real de uma farmácia.

## Referencias Bibliográficas.

COSTA, F. H. S.; SILVA, G. A. R.; MANHARELO, L.; CAMPOS, L. G. C.; MANTOVANI, V. H. Desenvolvimento de um sistema de gerenciamento de estoque: estudo de caso da empresa La Compagnie du Mobile. Orientador: Prof. Me. Tiago Filho Francisco da Costa. Faculdade de Tecnologia de Carapicuíba, 2018.

GARCIA, Letiene de Cássia; PEREIRA, Moacir; OSÓRIO, Wislei Riuper. GESTÃO DOS PARÂMETROS DE ESTOQUE: estudo de caso de itens de medicamentos em farmácias hospitalares e convencionais. Revista Gestão Industrial, [S.L.], v. 5, n. 1, p. 109-111, 1 jun. 2009. Universidade Tecnológica Federal do Paraná (UTFPR). <http://dx.doi.org/10.3895/s1808-04482009000100007>.

OLIVEIRA, M. A. *Sistema Gerenciamento de Drogaria*. Orientador: Alex Sandro Romeo de Souza Poletto. Instituto Municipal de Ensino Superior de Assis, 2008.

ROSA, Rodrigo Alvarenga. *Gestão e Operações e Logística*. 3. ed. Florianópolis: UFSC, 2015.

SILVA, A. G.; BERTAGLIA, G. C. R.; PERES, I. C.; BASÍLIO, E. R.; BASÍLIO, M. A.;

SILVA, N. E. F. *Sistema para Gerenciamento de Farmácia*. Orientador: Aduino Luís Michelotti Júnior. Etec Prof. Massuyuki Kawano, Tupã, 2015.

Compreendendo a arquitetura em multicamadas. Disponível em:  
<[https://www.macoratti.net/13/09/net\\_ncam.htm](https://www.macoratti.net/13/09/net_ncam.htm)>. Acesso em: 26 out. 2024.