****

**Projeto integrado: inovação**

**Faculdade Anhanguera – Osasco**

**Superior em Análise e desenvolvimento de sistemas**

**Tutor a distância: Luiz Gustavo Cardoso**

**Tutor presencial: Rita**

**Francisco Ismael Silva**

**OSASCO/2025**

**Relatório de Aula Prática**

**Situação Problema: Desenvolvimento de um Sistema de Gerenciamento de Estoque**

**Contexto:**

**Você foi contratado por uma empresa de comércio eletrônico que está em rápido crescimento. A Empresa enfrenta dificuldades para gerenciar seu estoque de forma eficiente, o que tem levado a**

**Problemas como falta de produtos, excesso de estoque e dificuldade em rastrear a localização dos Itens nos depósitos. A empresa deseja desenvolver um sistema de gerenciamento de estoque que Otimize suas operações, melhorando a eficiência e a precisão na gestão de produtos.**

**Problema:**

**Desenvolver um sistema de gerenciamento de estoque para a empresa de comércio eletrônico que Atenda aos seguintes requisitos:**

**• Cadastro de Produtos: O sistema deve permitir o cadastro de novos produtos, incluindo Informações como nome, categoria, quantidade em estoque, preço e localização no depósito.**

**• Atualização de Estoque: O sistema deve permitir a atualização da quantidade de produtos Em estoque quando novos itens são recebidos ou quando produtos são vendidos.**

**• Rastreamento de Localização: O sistema deve possibilitar o rastreamento da localização dos Produtos dentro dos depósitos para facilitar a logística.**

**• Relatórios: O sistema deve gerar relatórios sobre o estado do estoque, destacando produtos Com estoque baixo, excesso de estoque e movimentação de produtos.**

**• Tabela Verdade: Deve-se criar uma tabela verdade para definir as combinações de valores (True e False) para as variáveis booleanas da situação problema.**

**• Diagrama de Caso de Uso: Crie um Diagrama de Casos de Uso para o sistema de Gerenciamento de estoque da empresa.**

**Sumário**

[**1. Introdução** 4](#_Toc180175013)

[**2. Desenvolvimento** 5](#_Toc180175014)

[**3. Conclusão** 8](#_Toc180175016)

[**4. Referências** 9](#_Toc180175017)

# **Introdução**

Neste projeto, criei um Sistema de Gerenciamento de Estoque para ajudar no controle de produtos em estoque. O objetivo foi desenvolver uma solução prática para gerenciar as entradas e saídas de produtos, evitando problemas como falta ou excesso de estoque. Para isso, usei a metodologia ágil Scrum, dividindo o trabalho em três sprints: na primeira, foquei no cadastro de produtos; na segunda, implementei a atualização do estoque e relatórios; e, na terceira, fiz os ajustes finais e testes.

A Tabela Verdade foi usada para garantir que todas as funcionalidades do sistema estivessem funcionando corretamente. Ela verificou as combinações de condições, como a atualização de estoque e geração de relatórios, para garantir que o sistema atendesse a todos os requisitos.

O código do sistema foi desenvolvido para registrar produtos, atualizar o estoque e gerar relatórios de maneira simples e eficiente.

Além disso, criei um diagrama de casos de uso para mostrar como os usuários (Estoquista, Usuário e Gerente) interagem com o sistema. Isso ajudou a entender o papel de cada um e como as funções do sistema são executadas.

Esse trabalho me ajudou a aplicar meus conhecimentos de programação, organização de sistemas e análise de requisitos, e me mostrou como construir uma solução prática e eficiente.

# **2. Desenvolvimento**

Passo 1:

Sprint 1: Cadastro de Produtos e Configuração Inicial

Na primeira sprint, o foco foi a criação das funcionalidades básicas do sistema de gerenciamento de estoque. Essa etapa envolveu a implementação do cadastro de produtos, onde o usuário poderia adicionar itens ao estoque, incluindo informações como nome, categoria, quantidade e preço. A interação do Estoquista foi essencial nesta fase, pois ele seria o responsável por registrar a entrada dos produtos no sistema.

•Atividades principais da Sprint 1:

Cadastro de Produtos: Criamos a estrutura para cadastrar novos produtos no sistema. Cada produto tinha informações importantes como nome, categoria, quantidade em estoque, preço e localização no depósito.

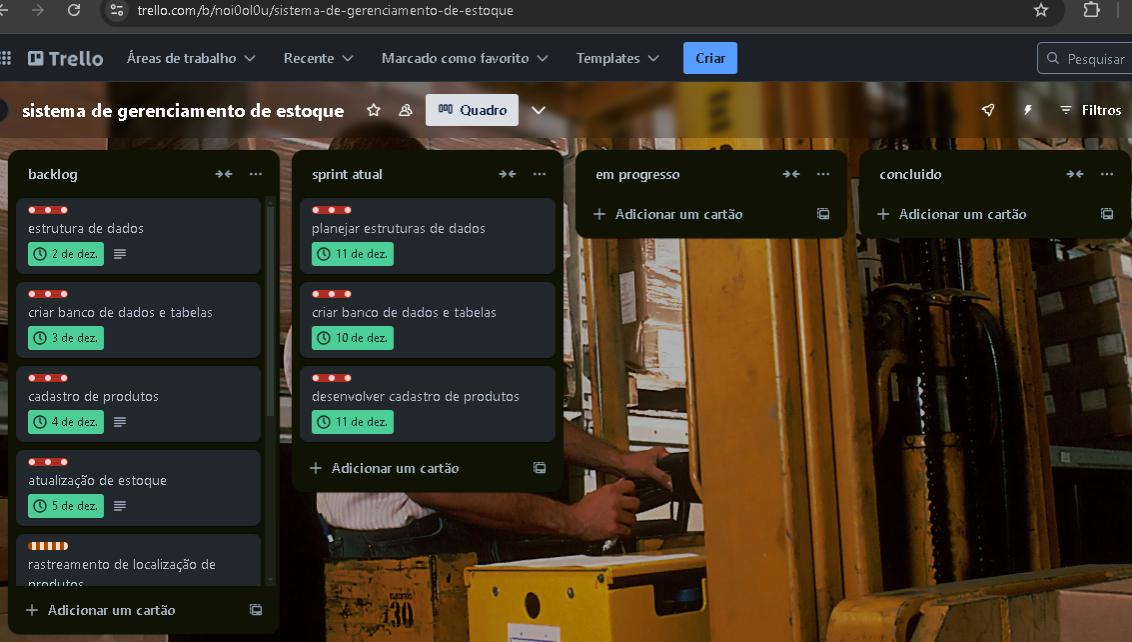
Configuração Inicial: Implementação de banco de dados para armazenar os dados dos produtos e garantir que o sistema fosse funcional.

Exemplo de Tarefas concluídas:

Criar banco de dados para os produtos.

Implementar tela de cadastro de produtos.

Validar informações de entrada dos produtos (preço, nome, quantidade).



Sprint 2: Atualização de Estoque e Relatórios

A segunda sprint teve como objetivo permitir a atualização do estoque e a geração de relatórios. O sistema agora permitia a modificação da quantidade de produtos no estoque, seja por entrada de novos produtos ou por saída de produtos vendidos. Além disso, implementamos uma funcionalidade para gerar relatórios, destacando os produtos com estoque baixo, excesso de estoque e outras movimentações.

•Atividades principais da Sprint 2:

Atualização de Estoque: Permitimos que o Estoquista registrasse as entradas e saídas de produtos.

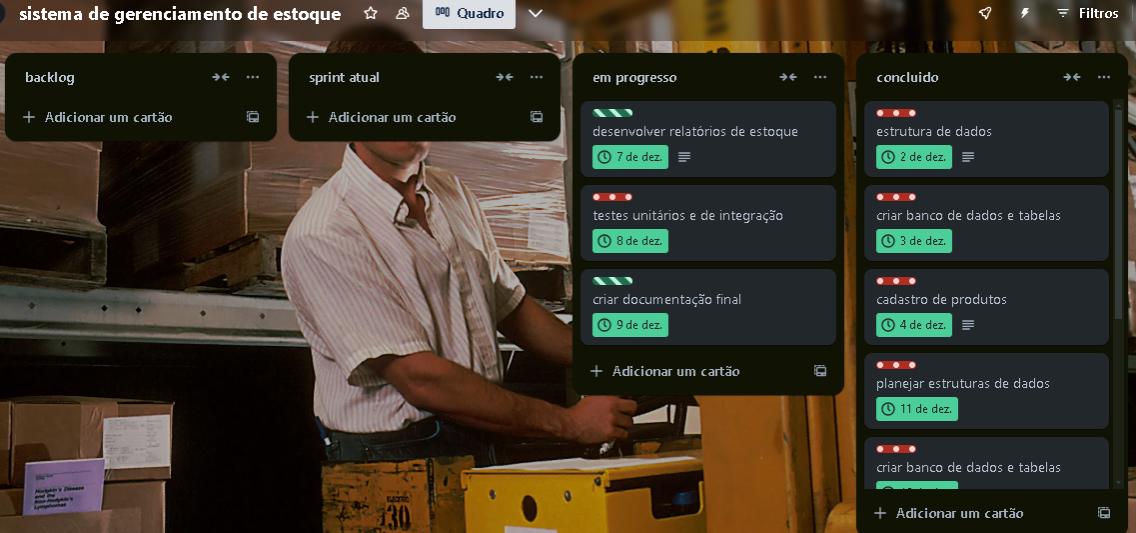
Geração de Relatórios: A funcionalidade de relatórios foi implementada para permitir a análise do estado do estoque em tempo real.

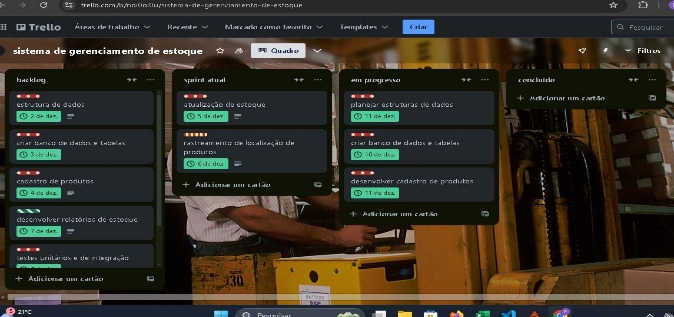
Exemplo de Tarefas concluídas:

Criar funcionalidades para atualizar a quantidade de estoque.

Implementar geração de relatórios de movimentação e estado do estoque.

Testar a integração entre as entradas de produtos e os relatórios gerados.





Sprint 3: Ajustes Finais e Testes

A última sprint foi dedicada aos ajustes finais e testes do sistema, garantindo que todas as funcionalidades estivessem funcionando corretamente. Durante esta fase, também revisamos o código, corrigimos erros e realizamos testes para garantir que o sistema fosse confiável e eficiente.

•Atividades principais da Sprint 3:

Testes do Sistema: Realizamos testes para verificar se as entradas de produtos e as atualizações de estoque estavam sendo registradas corretamente.

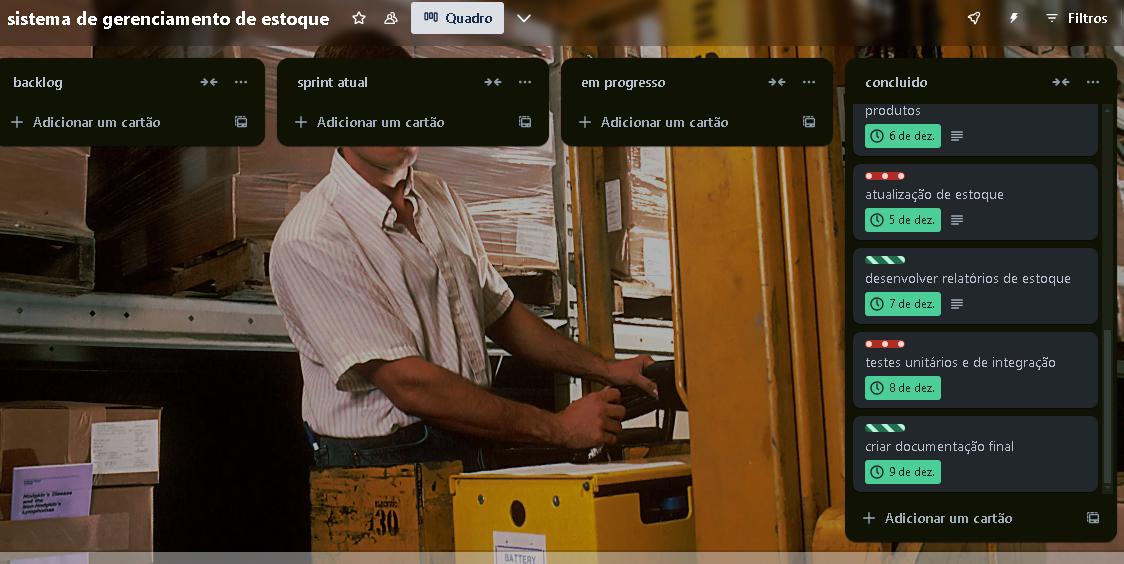
Ajustes Finais: Refinamos o código e corrigimos possíveis bugs.

Exemplo de Tarefas concluídas:

Testar a integridade dos dados no banco de dados.

Corrigir possíveis erros na atualização do estoque.

Validar o funcionamento da geração de relatórios em tempo real.

Com as três sprints finalizadas, conseguimos criar um Sistema de Gerenciamento de Estoque funcional, capaz de cadastrar produtos, atualizar o estoque e gerar relatórios de movimentação. Cada sprint teve um foco específico, e as tarefas foram divididas de forma eficiente para garantir que o sistema fosse desenvolvido de forma organizada e dentro do prazo.

Passo 2:

Tabela verdade

A tabela verdade é uma representação de todas as combinações possíveis de valores lógicos (Verdadeiro ou Falso) para as variáveis do problema. Neste caso, temos quatro variáveis booleanas que representam os requisitos do sistema:

P: Cadastro de Produtos

E: Atualização de Estoque

L: Rastreamento de Localização

R: Relatórios

A expressão lógica para que o sistema seja considerado completo é:

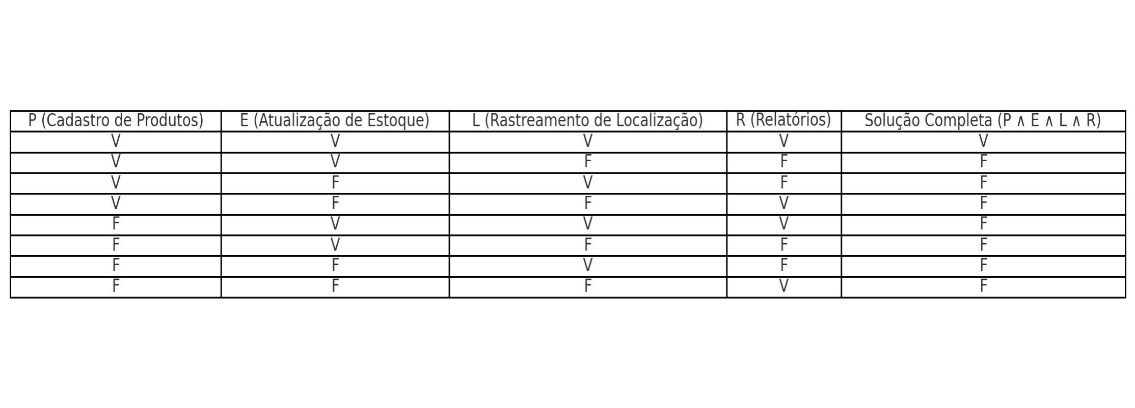
P ∧ E ∧ L ∧ R (AND lógico de todas as variáveis).

A solução será verdadeira (V) apenas quando todas as variáveis forem verdadeiras.

Explicação da Tabela:

Colunas de Variáveis: Representam os estados (Verdadeiro ou Falso) das quatro variáveis.

Solução Completa: É T (Verdadeiro) apenas quando P, E, L, e R são todos T simultaneamente.



Passo 3:

Criar as funções para definição das estruturas de dados, bem como algoritmos de cadastro e consulta Dos produtos, movimentações de estoque e relatórios.

**•Definição de Estruturas de Dados (em Python):**

# Estrutura para produtos

Class Produto:

Def \_\_init\_\_(self, nome, categoria, quantidade, preco, localizacao):

Self.nome = nome

Self.categoria = categoria

Self.quantidade = quantidade

Self.preco = preco

Self.localizacao = localizacao

**•Cadastro de Produtos:**

Def cadastrar\_produto(lista\_produtos):

Nome = input(“Nome do produto: “)

Categoria = input(“Categoria: “)

Quantidade = int(input(“Quantidade: “))

Preco = float(input(“Preço: “))

Localizacao = input(“Localização no depósito: “)

Produto = Produto(nome, categoria, quantidade, preco, localizacao)

Lista\_produtos.append(produto)

Print(f”Produto {nome} cadastrado com sucesso!”)

**•Atualização de Estoque:**

Def atualizar\_estoque(lista\_produtos, nome\_produto, quantidade):

For produto in lista\_produtos:

If produto.nome == nome\_produto:

Produto.quantidade += quantidade

Print(f”Estoque atualizado: {produto.quantidade} unidades de {produto.nome}”)

Return

Print(“Produto não encontrado!”)

**•Relatórios:**

Def gerar\_relatorio(lista\_produtos):

Print(“Relatório de Estoque:”)

For produto in lista\_produtos:

Status = “Baixo” if produto.quantidade < 10 else “Adequado”

Print(f”{produto.nome} – Quantidade: {produto.quantidade} – Status: {status}”)

No desenvolvimento do sistema de gerenciamento de estoque, focamos em criar soluções simples e eficazes para organizar e controlar os produtos da empresa. Utilizamos estruturas de dados como listas para armazenar informações sobre os produtos e seus estoques, o que facilita a consulta e a atualização. Além disso, criamos algoritmos que cuidam de tarefas como cadastrar novos produtos, atualizar o estoque quando há vendas ou entradas de mercadorias, e gerar relatórios para ajudar na tomada de decisões.

Com isso, o sistema foi estruturado de forma a ser fácil de usar, eficiente e capaz de atender às necessidades diárias de controle de estoque, trazendo mais organização e controle para a empresa.

Passo 4:

Diagrama de casos de uso

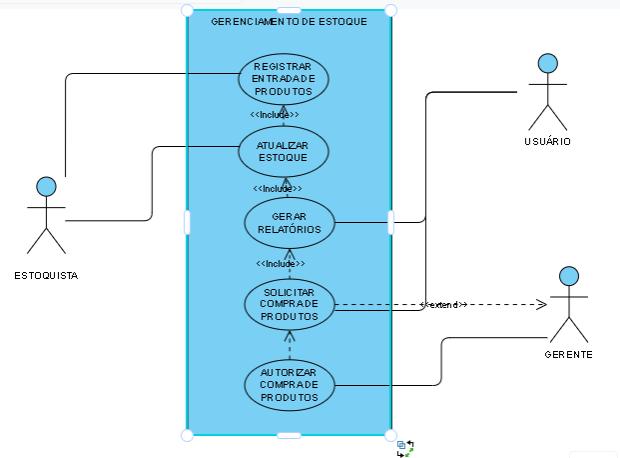
O diagrama de casos de uso do sistema de gerenciamento de estoque mostra de forma simples como as pessoas vão interagir com o sistema. Temos três tipos de usuários principais:

Estoquista: Ele é responsável por cadastrar novos produtos, atualizar o estoque e verificar onde os produtos estão no depósito.

Usuário: Esse ator pode consultar o estado do estoque e gerar relatórios sobre os produtos.

Gerente de Setor: O gerente autoriza compras e toma decisões com base nos relatórios do estoque.

O diagrama ajuda a entender claramente quais ações cada usuário pode fazer no sistema, garantindo que todas as funções necessárias estão no lugar e facilitando a gestão do estoque.

****

**3. Conclusão**

Esse projeto de gerenciamento de estoque foi uma ótima forma de aplicar e aprender vários conceitos. Usando o Trello, pude organizar as tarefas e acompanhar o progresso, o que ajudou a entender como planejar um projeto de forma eficiente. As estruturas de dados me ensinaram como armazenar e organizar informações de maneira prática.

A tabela verdade foi importante para garantir que o sistema funcionasse corretamente, e o diagrama de casos de uso me mostrou como estruturar as interações dos usuários com o sistema. Tudo isso me ajudou a entender como um sistema real é desenvolvido e a importância de cada etapa no processo.

Esse projeto foi valioso para meus estudos, pois me permitiu aplicar a teoria de forma prática, melhorando meus conhecimentos em desenvolvimento de sistemas, planejamento de projetos e lógica de programação.

# 

# **6. Referências**

**Trello: Ferramenta de gerenciamento de projetos que utiliza a metodologia ágil Scrum para organizar e acompanhar o progresso de tarefas e sprints.** Disponível em: [www.trello.com](http://www.trello.com).

**Estruturas de Dados: Conceito fundamental na programação que organiza e armazena dados de maneira eficiente, permitindo operações rápidas e fáceis de manipulação.** Fonte: Introdução à Ciência da Computação com Python.

**Tabela Verdade: Ferramenta da lógica booleana usada para validar as combinações possíveis de condições em um sistema.** Fonte: Matemática Discreta e Suas Aplicações.

**Diagrama de Casos de Uso: Técnica utilizada em Engenharia de Software para visualizar as interações entre os usuários (atores) e o sistema, representando as funcionalidades que o sistema deve oferecer.** Fonte: UML: Guia Prático para Análise e Projeto de Sistemas Orientados a Objetos.