

ANHANGUERA EDUCACIONAL

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Rafael dos Santos Gomes

**PROJETO INTEGRADO INOVAÇÃO - ADS**

Arcoverde

2024

RAFAEL DOS SANTOS GOMES

**PROJETO INTEGRADO DE INOVAÇÃO - ADS**

**ANHANGUERA EDUCACIONAL**

**ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

Trabalho referente ao Projeto Integrado Inovador acerca da situação problema: Desenvolvimento de um Sistema de Gerenciamento de Estoque.

ARCOVERDE

2024

**LISTA DE FIGURAS**

**Figura 1 -** Quadro Scrum 1 **05**

**Figura 2 -** Quadro Scrum 2 **05**

**Figura 3 -** Estrutura de dados **09**

**Figura 4 -** Cadastro **10**

**Figura 5 -** Consulta **11**

**Figura 6 -** Movimentação **11**

**Figura 7 -** Relatório de consulta **12**

**Figura 8 -** Teste **13**

**Figura 9 -** Resultado **14**

**Figura 10 -** Diagrama de caso de uso **15**

**LISTA DE TABELAS**

**Tabela 1 -** Tabela-Verdade **08**

**SUMÁRIO**

[**INTRODUÇÃO 5**](#_7oh4fnv4wcsn)

[**DESENVOLVIMENTO 5**](#_x2d22oi7mzde)

[Metodologia Ágil 5](#_bjeha3wpukhf)

[Tabela Verdade 7](#_az2rt5774e0g)

[Tabela 8](#_72ght9r6jlx7)

[Sistema de gerenciamento de estoque 8](#_ay7ldmp07bw4)

[Definição de Estrutura de Dados 9](#_k0ce2axjxebd)

[Algoritmos de Cadastro e Consulta 10](#_wma919lag6g8)

[Movimentação de Estoque 11](#_ds1dndf7ajng)

[Relatórios de Consultas 12](#_x9msm9j43ad3)

[Testes 13](#_glqdpod3i1ev)

[Diagrama de Caso de Uso 14](#_9u8ydjblrhje)

[Diagrama 15](#_xck9jmyb9rni)

[**CONCLUSÃO 16**](#_s95tnnq8t0b4)

[**REFERÊNCIAS 17**](#_57moj4ac3qcj)

[**ANEXOS 18**](#_yl960d9gx9re)

[**APÊNDICES 19**](#_43f8ook62z73)

# **INTRODUÇÃO**

O presente relatório aborda o desenvolvimento de um sistema de gerenciamento de estoque exigido pela empresa contratante, sendo essa focada no comércio eletrônico. O relatório detalha o problema apresentado e as etapas seguidas para sua solução de maneira clara e objetiva. Com este propósito, foi aplicada uma metodologia ágil, que proporcionou maior controle sobre as tarefas executadas. Além disso, foi elaborada uma tabela verdade para a verificação do atendimento aos requisitos estabelecidos, garantindo a conformidade do sistema com as expectativas. O desenvolvimento incluiu a criação de estruturas de dados e algoritmos capazes não apenas de gerenciar o estoque, mas também de fornecer informações detalhadas sobre sua situação. Por fim, um diagrama de caso de uso foi elaborado para ilustrar a interação dos diferentes atores com o sistema, tornando o processo mais transparente e compreensível.

# **DESENVOLVIMENTO**

## Metodologia Ágil

O primeiro passo foi analisar o problema, para assim escolher qual metodologia ágil seria melhor para abordar a situação. Observou-se que seria uma boa opção particionar o problema, tendo maior controle sobre cada etapa e respondendo rapidamente a imprevistos durante o desenvolvimento. Sendo assim, a metodologia escolhida foi a Scrum, por sua fácil aplicação e flexibilidade.

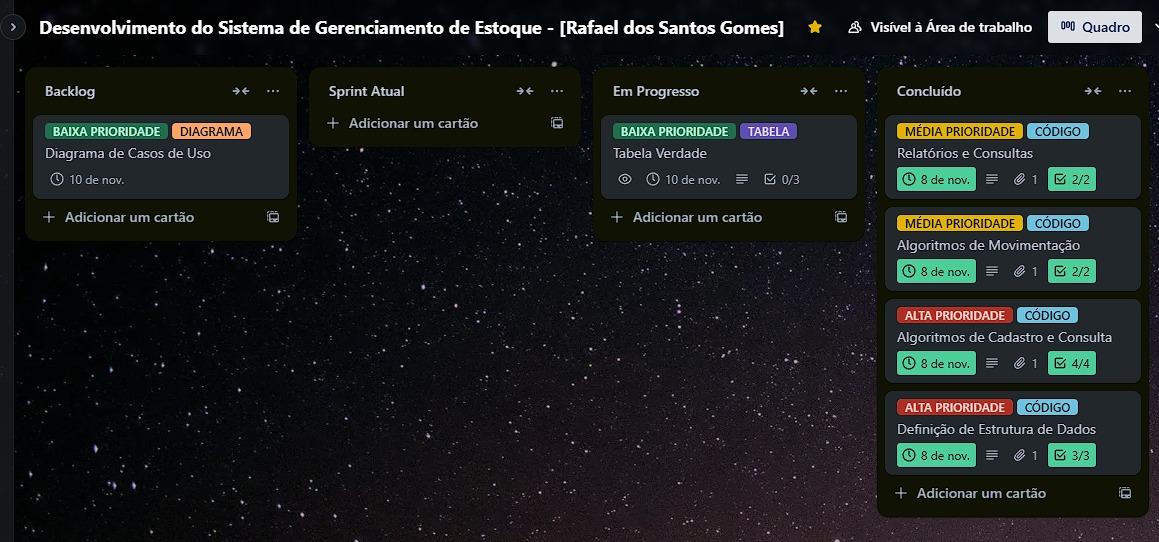
Para a aplicação do Scrum, foi utilizado a plataforma no site Trello, que se mostrou muito eficiente e dinâmica na aplicação da metodologia. Na plataforma, foi possível organizar as tarefas em seções, para definir cada etapa do desenvolvimento, as quais foram nomeadas como Backlog, Sprint atual, Em progresso e Concluído. Cada etapa tinha a sua função organizacional bem definida, para não haver erros de entendimento sobre o que deveria ser feito.

Para cada tarefa, foi atribuído seus níveis de prioridade, data de entrega, objetivos específicos além de anexos para os arquivos relacionados. A seguir, vemos capturas de tela (Prints) acerca do desenvolvimento do quadro Scrum:

**Figura 1**



Na figura 1 pode-se observar cada sessão citada anteriormente. No “Backlog” ficam todas as tarefas que serão feitas durante todo o desenvolvimento, sendo movidas para as demais seções conforme o desenvolvimento avança, e é também onde se define a prioridade e data de entrega, bem como os check lists de cada atividade. Já na “Sprint Atual”, se mantém as que serão feitas naquela sprint (instância de desenvolvimento) específica, priorizando as tarefas mais críticas do desenvolvimento. Se uma tarefa não for concluída em uma determinada sprint, ela pode ser alocada na próxima sprint, para que seja finalizada sem mais atraso.

**Figura 2**

Como visto na figura 2, na seção “Em Progresso”, como o próprio nome já diz, ficam as atividades que estão sendo feitas naquele momento, concluindo cada tópico do check list e sendo movidas para a próxima seção que é a “Concluído”. Nesta parte do processo, todos os requisitos de cada tarefa devem estar de acordo, para que o projeto seja apresentado ao cliente.

## Tabela Verdade

A análise de todas as possibilidades de atendimento de requisitos foi necessária. Com isso em mente, foi elaborada uma tabela-verdade, que por sua vez só retornaria o valor verdadeiro (True) se todas as variáveis fossem verdadeiras, caso contrário retornaria falso (False). Sendo assim, é possível deduzir que o operador lógico utilizado é o And (E ou /\).

As variáveis booleanas foram definidas da seguinte forma:

* P: Cadastro de Produtos;
* E: Atualização de Estoque;
* L: Rastreamento de Localização;
* R: Relatórios.

As condições para as variáveis retornarem o valor verdadeiro foram dadas de acordo com a interpretação abaixo:

* P (Produtos): Verdadeiro (V ou T) se o sistema permite o cadastro de novos produtos.
* E (Estoque): Verdadeiro (V ou T) se é permitido a atualização de produtos em estoque.
* L (Localização): Verdadeiro (V ou T) se é possível rastrear a localização dos produtos.
* R (Relatórios): Verdadeiro (V ou T) se relatórios são gerados sobre o estado do estoque.

Utilizando os símbolos lógicos para expressar o esquema, a seguinte interpretação foi montada:

P /\ E /\ L /\ R

Onde: Resultado será: P /\ E /\ L /\ R a operação lógica “E”, “/\” (AND).

### Tabela

**Tabela 1**

| **P** | **E** | **L** | **R** | **P /\ E** | **E /\ L** | **L /\ R** | **P /\ E /\ L /\ R** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| V | V | V | V | V | V | V | V |
| F | V | V | V | F | V | V | F |
| F | F | V | V | F | F | V | F |
| F | F | F | V | F | F | F | F |
| V | V | V | F | V | V | F | F |
| V | V | F | F | V | F | F | F |
| V | F | F | F | F | F | F | F |
| V | F | V | F | F | F | F | F |
| F | V | V | F | F | V | F | F |
| F | F | F | F | F | F | F | F |

A tabela 1 representa bem todas as possibilidades de atendimento dos requisitos, evidenciando que como o operador lógico foi o “E” (AND), é necessário que todas as variáveis sejam verdadeiras para que o resultado final também seja verdadeiro. Portanto, o software desenvolvido para o controle de estoque deverá atender a todos os requisitos para ser aprovado.

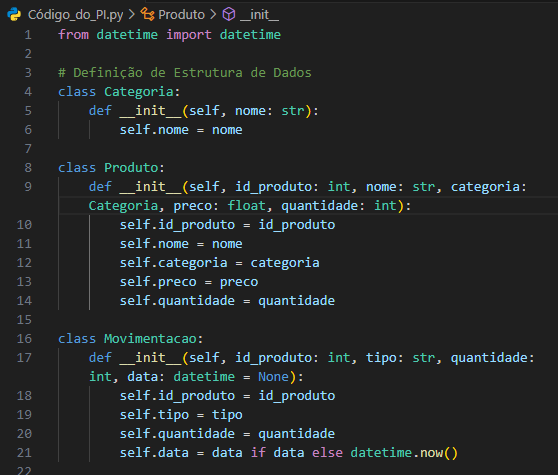
## Sistema de gerenciamento de estoque

De acordo com o problema da empresa em gerenciar o estoque, foi desenvolvido o sistema de gerenciamento de estoque. Porém, antes de começar o desenvolvimento do sistema, foi necessário escolher a linguagem de programação a ser usada. Pelo motivo de estar gerenciando dados e os manipulando, foi escolhida a linguagem Python, pois ela é amplamente utilizada para a análise e manipulação de dados. Além disso, também é muito intuitiva e clara em sua sintaxe, proporcionando assim um fácil entendimento do que cada linha de código faz.

O algoritmo foi separado em Definição de Estrutura de Dados, Algoritmos de Cadastro e Consulta, Algoritmos de Movimentação e por fim Relatórios e Consultas.

### Definição de Estrutura de Dados

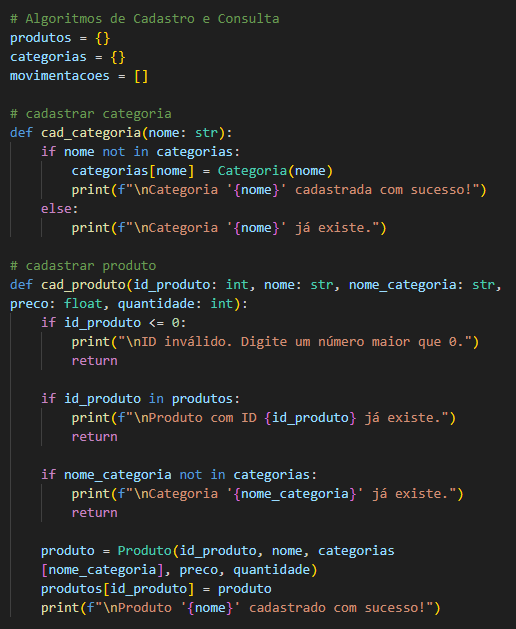
**Figura 3**



Primeiramente, foi importado o módulo datetime para a utilização da data atual do sistema operacional. Em seguida, foram criadas as classes Categoria, que serve de modelo para a criação de uma nova categoria, a classe Produto, que também é utilizada como base para a criação dos produtos e a classe Movimentação, que por sua vez é usada para criar dois tipos de movimentação, a de entrada e a de saída, além de verificar a data e hora da movimentação para um registro preciso da ação.

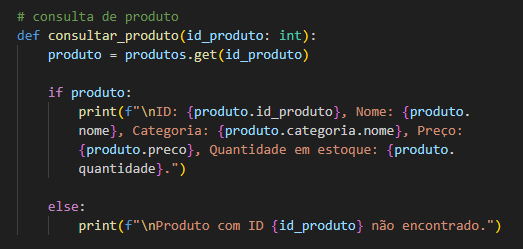
### Algoritmos de Cadastro e Consulta

**Figura 4**



Foram utilizados dicionários e listas para o armazenamento dos dados, tanto de produtos e categorias quanto das movimentações do estoque, além de funções de cadastro de categorias e produtos. Na função de cadastro de categoria, foram utilizadas estruturas condicionais para verificar se a categoria já existe ou não no sistema, se não, ele permite a criação da nova categoria. Também foi criada uma instância da classe “Categoria”, que possibilita a criação de uma nova categoria com base na própria estrutura. A função de cadastro de produtos, assim como a anterior, também utiliza estruturas condicionais e uma instância da classe “Produto”. Ela recebe atributos, que são informações desse produto e facilitam a manipulação dos dados. Os atributos são: id\_produto, nome, nome\_categoria, preco e quantidade.

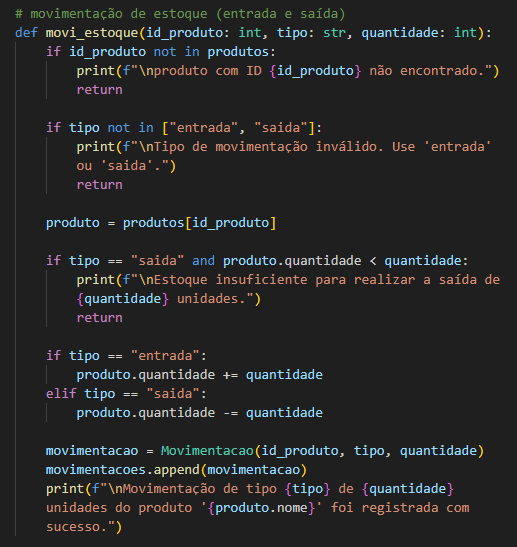
**Figura 5**



Para a consulta foi definida a função “consultar\_produto()” que percorre o dicionário “produtos" e mostra o item correspondente ao argumento referente ao ID do produto dado na função, mostrando assim as informações relacionadas a esse ID. Para isso é usada uma estrutura condicional para verificar se o produto está no dicionário ou não.

### Movimentação de Estoque

**Figura 6**

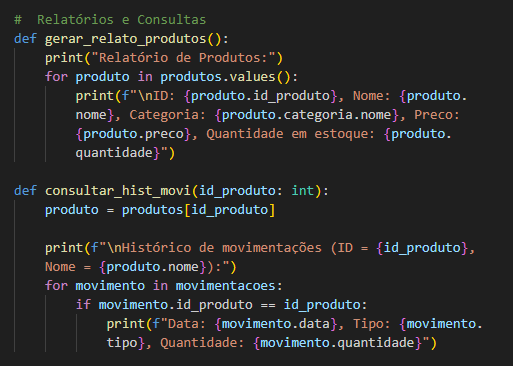


Para a movimentação do estoque foi elaborada a função “movi\_estoque()”, que recebe os argumentos “id\_produto”, “tipo” e “quantidade”, argumentos esses que serão usados para as movimentações de entrada e saída de produtos no estoque.

O argumento “tipo” define, através de estruturas condicionais, se o tipo de movimentação é de entrada ou de saída, deixando claro a movimentação realizada no estoque. Já o argumento “quantidade”, indica a quantidade que foi movimentada, seja ela de saída ou de entrada.

### Relatórios de Consultas

**Figura 7**



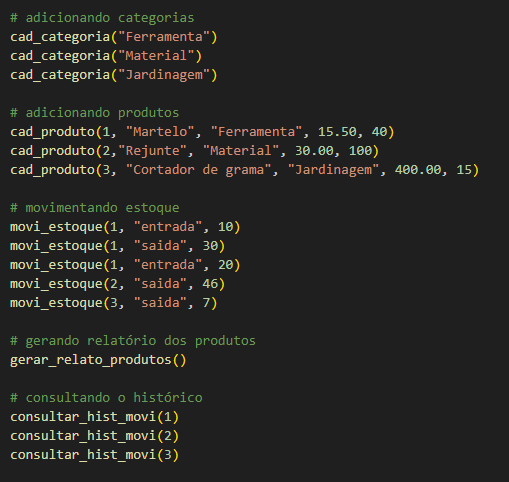
Para gerar os relatórios e fazer consultas, duas funções foram necessárias, sendo essas: as funções “gerar\_relato\_produtos()” e “consultar\_hist\_movi()”.

A função “gerar\_relato\_produtos()”, assim como seu nome indica, gera relatórios dos produtos em estoque, contendo informações como ID do produto, nome do produto, categoria em que está encaixado, preço, e quantidade em estoque.

Com o intuito de mostrar as movimentações realizadas no estoque, a função “consultar\_hit\_movi()” foi desenvolvida de forma que mostre o ID do produto, o nome, a data e hora da movimentação, o tipo, se foi entrada ou saída, e a quantidade movida.

### Testes

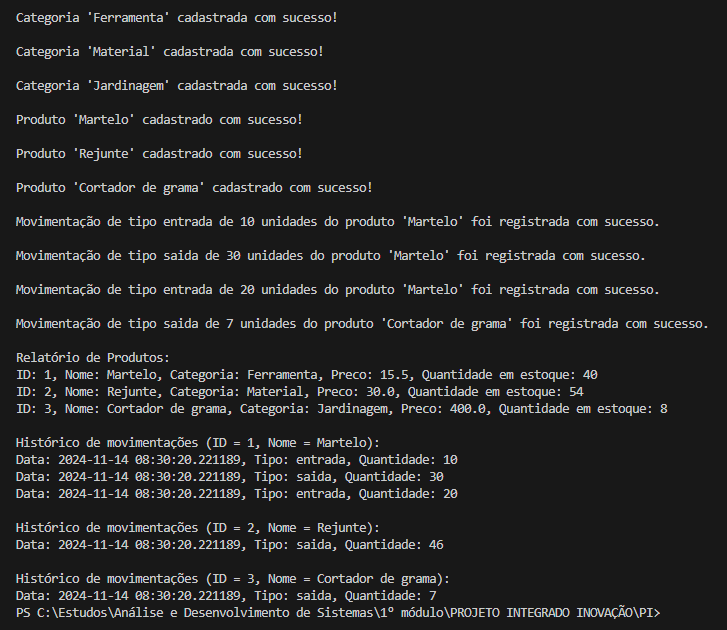
**Figura 8**



Para demonstrar o funcionamento e identificar erros no software, foram realizados testes breves de entrada e manipulação de dados, testando assim cada etapa do gerenciamento de estoque.

Foram adicionadas categorias usando a função “cad\_categoria()” para alocar os itens e manter o ambiente organizado. Os produtos foram cadastrados através da função “cad\_produto()”, inserindo suas devidas informações. Também foram realizadas as movimentações por meio da função “movi\_estoque()”, evidenciando a entrada e saída de produtos que já estavam em estoque e informando quantidade e o ID do item que seria movido. Para ter acesso ao relatório, a função “gerar\_relato\_produtos()” foi chamada, mostrando os dados referentes a cada produto. E, por fim, a consulta do histórico de movimentação de estoque foi realizada ao utilizar a função “consultar\_hist\_movi()”, que mostra todas as movimentações do produto solicitado por ID, indicando a data e hora, o tipo de movimentação e a quantidade movimentada.

**Figura 9**



O sistema respondeu como esperado e atendeu aos requisitos exigidos pelo cliente, concluindo assim a etapa de desenvolvimento do software.

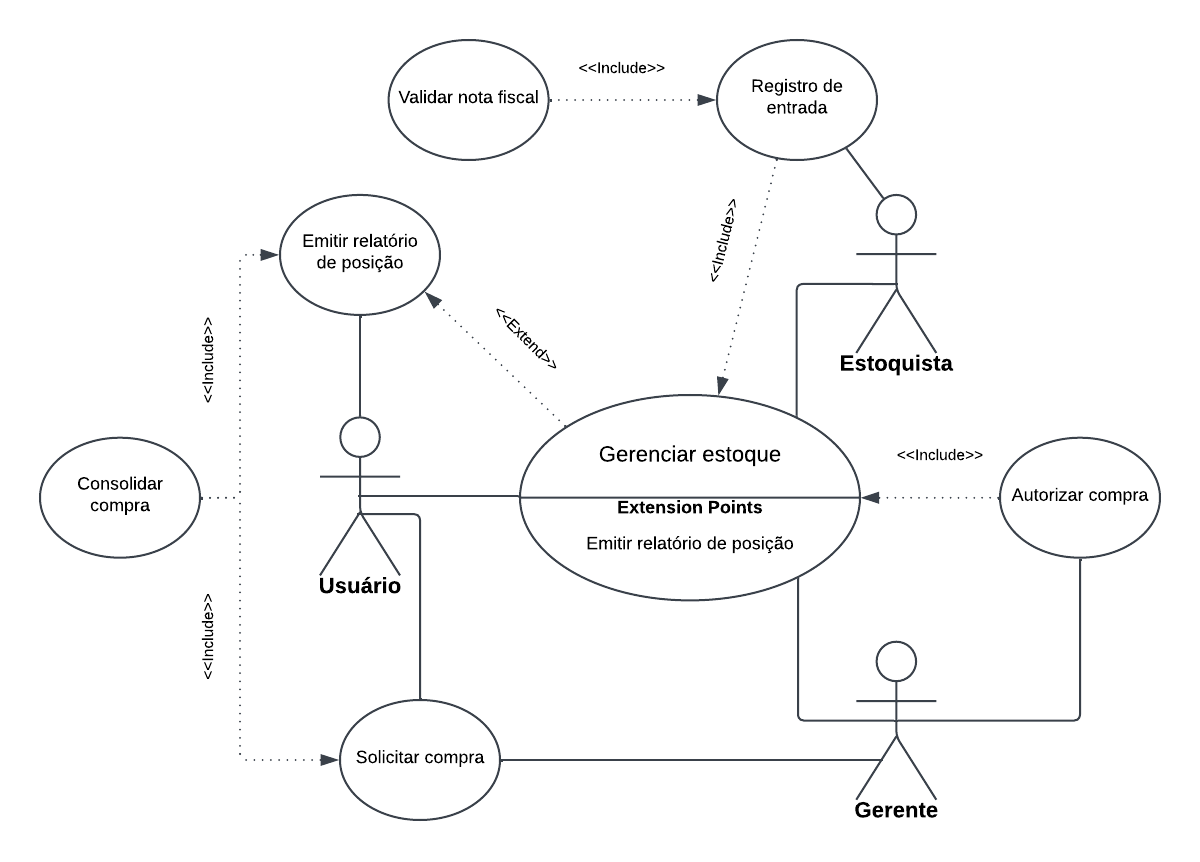
## Diagrama de Caso de Uso

Ser claro e objetivo é uma parte importante, principalmente ao apresentar o projeto ao cliente. Com isso em mente, foi elaborado um diagrama de caso de uso, no qual exemplifica a relação de cada ator envolvido com o gerenciamento de estoque.

A plataforma utilizada para o desenvolvimento do diagrama foi o site Lucid, que por sua vez, se mostrou eficiente e fácil de compreensão e usabilidade, além de ter um plano gratuito. Na plataforma, foi possível manter as informações bem organizadas e compactas, o que facilitou o desenvolvimento do diagrama.

### Diagrama

**Figura 10**



O principal caso de uso é o “Gerenciar estoque”, que é onde acontecem as saídas e entradas de produtos da empresa. A importância é evidente, visto que todos os atores e suas competências estão ligadas a ele. Também observa-se que nele estão incluídos os “Extension Points”, que são locais em um caso de uso onde funcionalidades adicionais podem ser incorporadas, mas de forma opcional e sob certas condições.

Os atores envolvidos no caso são o Usuário, que, nada mais é que o cliente que deseja realizar uma compra, o Estoquista, que controla as entradas dos produtos, e o Gerente, que aprova ou não a compra.

No lado do Usuário, ao realizar uma compra, ele faz com que o item que foi escolhido seja retirado do estoque, por isso ele tem relação. Para isso ele deve solicitar a compra e obrigatoriamente consolidá-la para que o pedido seja efetivado. Ele também pode emitir relatórios de posição, que também necessita da consolidação da compra.

O Estoquista, por ter controle direto do que entra no estoque, ele deve registrar essas ações, e para isso ele faz o registro de entrada, que necessita da validação da nota fiscal.

Por outro lado, o Gerente, como se pode deduzir, supervisiona o processo de controle do estoque. Ele tem mais responsabilidades, já que, para algum produto sair do estoque, é preciso ser aprovado por ele antes.

# **CONCLUSÃO**

Após a abordagem de todo o conteúdo e sua exemplificação de maneira detalhada, conclui-se que o desenvolvimento do sistema de gerenciamento de estoque foi bem-sucedido, pois está fundamentado na cobertura completa dos requisitos estabelecidos pelo contratante e em concordância também com a tabela-verdade que especifica a necessidade de que cada requisito seja contemplado.

A utilização da metodologia ágil Scrum foi essencial para um bom ordenamento das tarefas, o que facilitou o desenvolvimento geral, reforçando assim a eficiência das metodologias ágeis. Contudo, foram encontradas algumas dificuldades, como a decisão de quais atividades precisavam de mais atenção e como desenvolvê-las de maneira rápida e correta.

A importância deste relatório é evidente, visto que através dele pôde-se deixar clara cada etapa abordada, possibilitando assim a compreensão do sistema como um todo. Essa documentação não é apenas valiosa para a empresa contratante, mas também para futuros colaboradores e desenvolvedores que possam vir a trabalhar com o sistema. Ter um registro detalhado dos passos tomados e das soluções adotadas contribui para a manutenção e expansão do sistema de maneira mais eficiente e eficaz.

# **REFERÊNCIAS**

* **GUIA PARA ELABORAÇÃO DE TRABALHOS ACADÊMICOS.** USP. Disponível em: <https://usp.br/sddarquivos/arquivos/abnt14724\_2022.pdf>. Acesso em 06 de nov. de 2024.
* **O que é o Scrum?.** AWS. Disponível em: <https://aws.amazon.com/pt/what-is/scrum/#:~:text=O%20Scrum%20%C3%A9%20um%20framework,uma%20entrega%20eficiente%20de%20projetos.>. Acesso em 06 de nov. de 2024.
* **LÓGICA-PROPOSIÇÕES-CONECTIVOS-TABELA-VERDADE.** INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO RIO GRANDE DO NORTE. Disponível em: <https://docente.ifrn.edu.br/cleonelima/disciplinas/fundamentos-de-programacao-2.8401.1m/fundamentos-de-logica-e-algoritmos-1.8401.1v/apostila-proposicoes-tabelas-verdade-conectivos-logicos>. Acesso em 08 de nov. de 2024.
* **Documentação Python 3.14.0a1.** Python. Disponível em: <https://docs.python.org/pt-br/3.14/>. Acesso em 03 de nov. de 2024.
* **Diagramas de Caso de Uso.** IBM. Disponível em: <https://www.ibm.com/docs/pt-br/rsm/7.5.0?topic=diagrams-use-case>. Acesso em 10 de nov. de 2024.

# 

# **APÊNDICES**

**APÊNDICE A - Sistema de Gerenciamento de Estoque.** Disponível em: <https://github.com/Rafa57/PI/blob/main/Script(PI).py>

**APÊNDICE B - Quadro Scrum.** Disponível em**:** <<https://trello.com/b/IRXGBPpO/desenvolvimento-do-sistema-de-gerenciamento-de-estoque-rafael-dos-santos-gomes>>

**APÊNDICE C - DIAGRAMA DE CASO DE USO.** Disponível em**:** <<https://lucid.app/lucidchart/dde9223c-6526-439a-8f18-15534bd679f8/edit?page=0_0&invitationId=inv_3ae34a6d-0ac9-440f-bd2a-c07932500d6a#>>