

RELATÓRIO PROJETO N2

Sistemas de Informação – 01J12

Eduardo Hiroyuki Tamaributi – 32331762

Julia Kovacs Takamura – 32371489

Decisões de implementação

Para a implementação do projeto, foram criadas 5 funções que executam as tarefas de cadastrar, consultar, atualizar, excluir e ver um relatório completo do armazém.

Os produtos foram criados na forma de dicionário, contendo um nome, um código e sua quantidade. Depois do item ser cadastrado, ele é adicionado à lista estoque.

A função de consultar um produto itera sobre os itens da lista estoque comparando o código do produto com o código fornecido pelo usuário, caso o produto seja encontrado, a função exibe as informações do produto, caso contrário, uma mensagem de erro será exibida.

A função de atualizar as informações de um produto utiliza o mesmo método citado previamente para identificar se o código solicitado pelo usuário está presente na lista de produtos, caso estiver, é perguntado ao usuário se ele deseja atualizar o nome e/ou a quantidade daquele determinado produto.

A função de excluir um produto também utiliza o mesmo método para identificar se o código solicitado está presente na lista de produtos, caso esteja, é perguntado se o usuário deseja remover o item selecionado.

A função de exibir um relatório do armazém primeiramente exibe um cabeçalho com os tópicos: nome, código e quantidade. Além disso, é feito um cálculo de espaçamento para a formatação da tabela, ou seja, a coluna de nomes irá possuir a largura do produto de maior nome. Depois de exibir o cabeçalho, as informações de cada produto são exibidas em cada linha da tabela.

[Planejamento no Figma](#)

Printscreen com testes de execução de todas as opções do Menu

CADASTRAR PRODUTO	<div>Opção escolhida: 1 Nome do produto: Batata Código do produto: 1234 Quant: 100 Batata cadastrado com sucesso</div>
CONSULTAR PRODUTO	<div>Opção escolhida: 2 Código do produto: 1234 Produto Codigo Quant Batata 1234 100</div>
ATUALIZAR PRODUTO	<div>Opção escolhida: 3 Código do produto: 1234 Produto a ser atualizado: Batata Deseja atualizar o nome (s/n): s Nome atualizado: Cenoura Deseja atualizar a quant (s/n): s Quant atualizada: 50 Cenoura atualizado com sucesso Produto Codigo Quant Cenoura 1234 50</div>
REMOVER PRODUTO	<div>Opção escolhida: 4 Código do produto: 1234 Deseja remover o item Cenoura (s/n): s Produto removido com sucesso</div>
RELATÓRIO DE PRODUTOS	<div>Opção escolhida: 5 Produto Codigo Quant Abobrinha 5555 200 Cenoura 1234 100</div>

Código Final Desenvolvido

```
def insira_codigo () :  
    # Função para o usuário inserir o código do produto | Como o  
    código foi utilizado diversas vezes, ele foi transformado em uma  
    função  
    while True :  
        # Recebe um input do código do produto e caso ele possuir 4  
        dígitos, retorna o valor  
        codigo = int(input("Código do produto: "))  
        if codigo > 0 and codigo < 10000 :  
            return codigo  
        else :  
            print('Erro! Código deve conter no máximo 4 dígitos\n')  
def cadastra_produto () :  
    # Cria um dicionário {produto} que possui as categorias: none,  
    código e quantidade  
    produto = {}  
    produto['nome'] = input("Nome do produto: ")  
    produto['code'] = insira_codigo()  
    while True :  
        produto['quant'] = int(input("Quant: "))  
        if produto['quant'] > 0 :  
            break  
        else :  
            print("Erro! Quant deve ser maior que 0\n")  
    print(f'{produto["nome"]} cadastrado com sucesso\n')  
    return produto  
def consulta_produto (lista, codigo) :  
    # Faz a consulta de um produto específico cadastrado  
    for i in range (len(lista)) : # Itera entre os elementos da lista  
    [estoque]  
        if lista[i]['code'] == codigo :  
            # Caso o código do elemento for o mesmo que o parâmetro da  
            função, printa esse elemento  
            espacamento = len(lista[i]['nome'])  
            if espacamento < 7 :  
                espacamento = 7  
            print(f'Produto' + ' ' * (espacamento - 7) + '| Código |  
Quant')
```

```

        print(f'{lista[i]["nome"]}' + ' ' * (espacamento -
len(lista[i]['nome'])) + f'| {lista[i]["code"]:4.0f} |
{lista[i]["quant"]}\n')
        break
    else :
        # Caso não haja correspondência para o código inserido como
parâmetro, devolve uma mensagem de erro
        print("Erro! Produto não encontrado\n")
def atualiza_produto (lista, codigo) :
    # Atualiza o nome e quantidade de um produto da lista [estoque]
    for i in range (len(lista)) : # Itera entre os elementos da lista
[estoque]
        if lista[i]['code']== codigo :
            # Caso o código do elemento for o mesmo que o parâmetro da
função, atualiza o seu nome e/ou quantidade
            print(f'Produto a ser atualizado: {lista[i]["nome"]}')
            while True :
                att_nome = input("Deseja atualizar o nome (s/n): ")
                if att_nome.lower() == 's' :
                    lista[i]['nome'] = input("Nome atualizado: ")
                    break
                elif att_nome.lower() == 'n' :
                    break
                else :
                    print("Erro! Comando inválido\n")
            while True :
                att_quant = input("Deseja atualizar a quant (s/n): ")
                if att_quant.lower() == 's' :
                    lista[i]['quant'] = int(input("Quant atualizada:
"))
                    break
                elif att_quant.lower() == 'n' :
                    break
                else :
                    print("Erro! Comando inválido\n")
            print(f'{lista[i]["nome"]} atualizado com sucesso\n')
            consulta_produto(lista, codigo)
            break
    else :

```

```

        # Caso não haja correspondência para o código inserido como
parâmetro, devolve uma mensagem de erro
        print("Erro! Produto não encontrado\n")
def excluir_produto (lista, codigo) :
    # Exclui um determinado elemento da lista [estoque]
    for i in range (len(lista)) : # Itera entre os elementos da lista
[estoque]
        if lista[i]['code'] == codigo :
            # Caso o código do elemento for o mesmo que o parâmetro da
função, pergunta se o usuário quer fazer a remoção do item
            while True :
                confirmar = input(f"Deseja remover o item
{lista[i]['nome']} (s/n): ")
                if confirmar == 's' :
                    lista.pop(i)
                    print("Produto removido com sucesso\n")
                    break
                elif confirmar == 'n' :
                    print("Operação cancelada\n")
                    break
                else :
                    print('Erro! Comando inválido')
            break
        else :
            print("Erro! Produto não encontrado\n")
def exibe_relatorio (lista) :
    # Exibe um relatório do estoque apresentando o nome, código e
quantidade de cada item
    if len(lista) > 0 :
        # Faz o cabeçalho do relatório
        lista = sorted(lista, key=lambda d: d['nome'])
        espacamento = len(lista[0]['nome'])
        for i in range (len(lista)) :
            if len(lista[i]['nome']) > espacamento :
                espacamento = len(lista[i]['nome'])
        espacamento -= 7
        if espacamento < 0 :
            espacamento = 0
        print('Produto', end='')
        print(' ' * espacamento, end='|')

```

```

        print('Codigo | Quant')
        espacamento += 7
        for i in range (len(lista)) :
            # Printa os itens
            print(lista[i]['nome'], end='')
            print(' ' * (espacamento - len(lista[i]['nome'])),
end='|')

            print(f' {lista[i]["code"]:4.0f} |
{lista[i]["quant"]}')
            print()
        else :
            # Caso a lista [estoque] esteja vazia, retorna uma mensagem de
erro

            print("Erro! Armazém vazio\n")

print('+++++++ MENU - CONTROLE DE ESTOQUE ++++++')
estoque = []
while True :
    # Printa as opções que o usuário possui
    print('1. Cadastrar Produto\n2. Consultar Produto\n3. Atualizar
Produto\n4. Excluir Produto\n5. Relatório de Produtos\n6. Encerrar')
    escolha = int(input("Opção escolhida: "))
    print()
    if escolha == 6 :
        break
    elif escolha == 1 :
        estoque.append(cadastra_produto())
    elif escolha == 2 :
        consulta_produto(estoque, insira_codigo())
    elif escolha == 3 :
        atualiza_produto(estoque, insira_codigo())
    elif escolha == 4 :
        excluir_produto(estoque, insira_codigo())
    elif escolha == 5 :
        exibe_relatorio(estoque)
    else :
        print("Erro! Comando inválido\n")

```