

Narrativa de Casos de Uso

Caso de Uso 01 – Adicionar Item ao Estoque

Identificação: CU01

Nome: Adicionar Item (/add)

Atores: Usuário, Bot

Objetivo: Adicionar um novo item ao estoque ou aumentar a quantidade de um item existente.

Pré-condição: O bot deve ser mencionado na mensagem.

Pós-condição: O item é salvo no inventário do chat e gravado arquivo json

Fluxo Principal

1. O Usuário menciona o bot e envia o comando /add <item> <quantidade>.
2. O Bot verifica se a quantidade é válida (número inteiro positivo).
3. O Bot adiciona o item ao inventário do chat ou incrementa sua quantidade.
4. O Bot salva a alteração no arquivo de persistência.
5. O Bot retorna uma mensagem confirmando:
“Adicionou X unidades de Y.”

Fluxo Alternativo A – Quantidade inválida

A1. O usuário envia quantidade negativa ou não numérica.

A2. O bot responde: “Quantidade inválida. Use apenas números positivos.”

Caso de Uso 02 – Subtrair Quantidade de um Item

Identificação: CU02

Nome: Subtrair Item (/sub)

Atores: Usuário, Bot

Objetivo: Diminuir a quantidade de um item do estoque.

Pré-condição: O item deve existir no inventário e a subtração não pode gerar valor negativo.

Pós-condição: O estoque é atualizado e salvo no arquivo.

Fluxo Principal

1. O Usuário menciona o bot e envia /sub <item> <quantidade>.
2. O Bot verifica se o item existe.
3. O Bot valida se a quantidade é válida e menor ou igual à quantidade atual.
4. O Bot subtrai o valor informado.
5. O Bot salva a alteração.
6. O Bot envia mensagem: “Subtraiu X unidades de Y.”

Fluxo Alternativo A – Item inexistente

A1. O bot responde: “Item não encontrado no estoque.”

Fluxo Alternativo B – Quantidade maior que a disponível

B1. O bot responde:

“Não é possível subtrair essa quantidade. O valor ficaria negativo.”

Caso de Uso 03 – Definir Quantidade Exata

Identificação: CU03

Nome: Definir Quantidade (/set)

Atores: Usuário, Bot

Objetivo: Definir uma quantidade exata de um item no estoque.

Pré-condição: O item deve existir no inventário.

Pós-condição: A nova quantidade é atualizada e salva.

Fluxo Principal

1. O Usuário envia /set <item> <quantidade> mencionando o bot.
2. O Bot verifica se o item existe no inventário.
3. O Bot valida se a quantidade é apropriada (inteiro positivo ou zero).
4. O Bot atualiza a quantidade exata.
5. O Bot salva no arquivo.
6. O Bot responde: “Agora Y possui X unidades.”

Fluxo Alternativo A – Item inexistente

A1. O bot responde: “Item não encontrado no estoque.”

Caso de Uso 04 – Remover Item

Identificação: CU04

Nome: Remover Item (/remove)

Atores: Usuário, Bot

Objetivo: Excluir completamente um item do inventário.

Pré-condição: O item deve existir no inventário.

Pós-condição: O item é removido e o inventário atualizado.

Fluxo Principal

1. O Usuário menciona o bot e envia /remove <item>.
2. O Bot verifica se o item existe.
3. O Bot remove o item do inventário.
4. O Bot salva a alteração no arquivo.
5. O Bot responde: “Item X removido com sucesso.”

Fluxo Alternativo A – Item inexistente

A1. O bot responde: “Item não encontrado.”

Caso de Uso 05 – Listar Itens do Estoque

Identificação: CU05

Nome: Listar Estoque (/list)

Atores: Usuário, Bot

Objetivo: Mostrar todos os itens e suas quantidades no chat atual.

Pré-condição: O inventário pode estar vazio.

Pós-condição: O usuário visualiza o estoque atualizado.

Fluxo Principal

1. O usuário menciona o bot e envia /list.
2. O Bot acessa o inventário correspondente ao chat.
3. O Bot monta a lista de itens e quantidades.
4. O Bot responde exibindo todo o estoque.

Fluxo Alternativo A – Estoque vazio

A1. O bot responde: “O estoque está vazio no momento.”

Caso de Uso 06 – Processar Apenas Quando Mencionado

Identificação: CU06

Nome: Processar Mensagem com Menção

Atores: Usuário, Bot

Objetivo: Garantir que o bot só processe comandos quando é explicitamente chamado.

Pré-condição: —

Pós-condição: Comandos sem menção são ignorados.

Fluxo Principal

1. O Bot recebe uma mensagem no chat.
2. O Bot verifica se o texto contém seu nome (ex.: “@Duds_bot”).
3. Se houver menção, o Bot processa o comando normalmente.
4. Se não houver, o Bot ignora a mensagem.

Caso de Uso 07 – Persistir Dados

Identificação: CU07

Nome: Salvar Alterações no Arquivo

Atores: Bot

Objetivo: Garantir que o inventário seja mantido mesmo após reiniciar o sistema.

Pré-condição: Uma operação de modificação deve ocorrer.

Pós-condição: O arquivo json é atualizado.

Fluxo Principal

1. O Bot executa uma operação que altera o inventário (/add, /sub, /set, /remove).
2. O Bot abre ou acessa o arquivo json.
3. O Bot grava o inventário atualizado.
4. O Bot mantém os dados persistentes mesmo em reinicializações.