

# Documentação: Aplicação de Gerenciamento de Estoque com RMI

## 1. Visão Geral

A aplicação de gerenciamento de estoque com RMI (Remote Method Invocation) permite adicionar produtos ao estoque e visualizar o estoque atual. A aplicação é dividida em duas partes principais: o cliente e o servidor RMI.

O cliente fornece uma interface gráfica simples para adicionar produtos ao estoque e visualizar o estoque atual. Ele se comunica com o servidor RMI para executar essas operações.

O servidor RMI é responsável por armazenar os produtos e suas quantidades. Ele implementa uma interface remota que define os métodos disponíveis para adicionar produtos e obter o estoque.

## 2. Classes Principais

A aplicação é composta por várias classes, mas as principais são:

### 2.1. StockRMIClient

- Descrição: Classe responsável pela interface gráfica do cliente e pela comunicação com o servidor RMI.
- Métodos principais:
  - **'main(String[] args)'**: O método principal que inicializa a aplicação cliente.
  - **'addProduct()'**: Método invocado quando o botão "Add Product" é clicado. Obtém o nome do produto e a quantidade do estoque a partir dos campos de texto e chama o método **'addProduct'** do servidor RMI para adicionar o produto ao estoque.
  - **'viewStock()'**: Método invocado quando o botão "View Stock" é clicado. Chama o método **'getStock'** do servidor RMI para obter o estoque atual e exibe as informações na área de texto.

### 2.2. Stock

- Descrição: Classe responsável pela implementação do servidor RMI e pelo armazenamento dos produtos e quantidades.
- Métodos principais:

- **'addProduct(String productName, int quantity)'**: Adiciona um produto ao estoque com o nome e a quantidade fornecidos.
- **'getStock()'**: Retorna um mapa contendo o estoque atual, onde as chaves são os nomes dos produtos e os valores são as quantidades.

## 2.3. StockRMI

- Descrição: Interface remota que define os métodos disponíveis para o cliente se comunicar com o servidor RMI.
- Métodos principais:
  - **'addProduct(String productName, int quantity)'**: Adiciona um produto ao estoque com o nome e a quantidade fornecidos.
  - **'getStock()'**: Retorna um mapa contendo o estoque atual.

## 3. Funcionamento da Aplicação

- 3.1. O servidor RMI é iniciado e aguarda as solicitações do cliente.
- 3.2. O cliente é iniciado e exibe uma janela com campos de texto para inserir o nome do produto e a quantidade do estoque, bem como botões para adicionar produtos e visualizar o estoque.
- 3.3. Quando o usuário clica no botão "Add Product", o cliente obtém o nome do produto e a quantidade do estoque dos campos de texto e chama o método **'addProduct'** do servidor RMI para adicionar o produto ao estoque.
- 3.4. Quando o usuário clica no botão "View Stock", o cliente chama o método **'getStock'** do servidor RMI para obter o estoque atual e exibe as informações na área de texto.
- 3.5. O servidor RMI processa as solicitações do cliente, atualiza o estoque conforme necessário e retorna os resultados ao cliente.
- 3.6. O cliente exibe as informações atualizadas na interface gráfica.

## 4. Configuração do Ambiente

Para executar a aplicação corretamente, siga estas etapas:

- 4.1. Compile todas as classes Java do projeto.
- 4.2. Execute o registro RMI para disponibilizar o serviço do servidor RMI. Você pode usar o comando **'rmiregistry'** em um terminal.
- 4.3. Inicie o servidor RMI chamando o método **'rebind'** no objeto remoto do servidor. Certifique-se de que o servidor RMI esteja sendo iniciado corretamente e esteja vinculado ao registro RMI.
- 4.4. Inicie o cliente chamando o método **'main'** na classe **'StockRMIClient'**. Certifique-se de que o cliente esteja se conectando ao registro RMI correto e possa localizar o servidor RMI.
- 4.5. A interface gráfica do cliente será exibida. Você pode adicionar produtos ao estoque fornecendo o nome e a quantidade nos campos de texto e clicando

no botão "Add Product". Para visualizar o estoque atual, clique no botão "View Stock".

## **5. Considerações Finais**

A documentação fornecida abrange os aspectos básicos da aplicação de gerenciamento de estoque com RMI. No entanto, é possível aprimorar a aplicação implementando recursos adicionais, como remoção de produtos do estoque, atualização de quantidades, autenticação de usuários, entre outros.