

Projeto: Sistema de Gestão de Livraria Online

Visão Geral:

O projeto envolve o desenvolvimento de um sistema de gestão de livraria online onde os usuários podem navegar por livros, fazer pedidos e gerenciar suas contas. O sistema consiste em múltiplos componentes que se comunicam por meio de RPC.

Componentes e Uso de RPC:

1. Aplicação Cliente:

- Os usuários interagem com a livraria por meio de um aplicativo web ou mobile.
- O aplicativo cliente comunica-se com os componentes do lado do servidor usando chamadas RPC.

2. Componentes do Servidor:

• Catálogo de Livros

- Conjunto de livros previamente cadastrados, para cada livro as seguintes informações: título, autor, ano (outras informações sobre o livro que considerar pertinente), quantidade de exemplares em estoque, preço.
- Expõe método RPC para consulta de informações de livros

• Serviço de Gestão de Pedidos:

- Lida com a realização de pedidos
- Fornece métodos RPC para realizar pedidos (selecionar através do título do livro), informando a quantidade, obter detalhes de um pedido específico (retorna informações sobre o pedido e data do pedido) e recuperar histórico de pedidos (conjunto de todos os pedidos realizados pelo usuário, contendo somente o título do livro).
- Um pedido pode apresentar vários itens (livros).
- Ao finalizar um pedido o cliente deve receber o ID desse pedido (identificador único).
- O ID será utilizado para obter detalhes sobre pedido.

• Serviço de Autenticação de Usuário:

- Gerencia contas de usuário e autenticação.
- Métodos RPC para login de usuário e registro de novos usuários

3. Framework RPC:

- Utilize gRPC para definir serviços e métodos.

Detalhes de Implementação:

• Comunicação Cliente-Servidor:

- Os clientes (web ou mobile) enviam solicitações RPC para o servidor para realizar operações como navegar por livros, adicionar itens ao carrinho (pedidos).

- O servidor processa essas solicitações, interage com bancos de dados ou APIs externas e envia respostas de volta aos clientes.
- **Arquitetura de Serviço:**
 - Cada serviço (Catálogo, Gestão de Pedidos, Autenticação) é um módulo separado com sua própria interface RPC definida em protobuf.
 - gRPC lida com a comunicação entre cliente e servidor, gerenciando pooling de conexões, balanceamento de carga e tratamento de erros.
- **Segurança e Escalabilidade:**
 - Implemente mecanismos de autenticação e autorização usando tokens ou gerenciamento de sessões.
 - Projete serviços para serem escaláveis, permitindo que o sistema lide com um número crescente de usuários e transações.

Fluxo de Trabalho Exemplo:

1. Registro e Login do Usuário:

- O cliente envia uma solicitação RPC ao Serviço de Autenticação para registrar um novo usuário ou autenticar um usuário existente.
- O servidor verifica as credenciais e retorna um token para sessões autenticadas.

2. Navegação e Pedido de Livros:

- O cliente solicita informações sobre livros do Serviço de Catálogo usando RPC.
- Os usuários adicionam livros ao carrinho e procedem para o checkout.
- Chamadas RPC são feitas ao Serviço de Gestão de Pedidos para realizar o pedido.

3. Rastreamento de Pedidos:

- Os clientes podem ver seu histórico de pedidos e detalhes sobre um determinado pedido. Para obter detalhes de um pedido específico, defina uma chave de busca, como por exemplo, um ID para cada pedido, esse ID pode ser obtido através do historico de pedidos ou recebido como resposta ao fechamento do pedido.
- Chamadas RPC recuperam detalhes do pedido e histórico de pedidos.

Benefícios do RPC neste Projeto:

- **Modularidade:** Serviços podem ser desenvolvidos, testados e escalados de forma independente.
- **Eficiência:** gRPC e Protocol Buffers garantem comunicação eficiente e serialização de dados.
- **Facilidade de Desenvolvimento:** RPC abstrai as complexidades de comunicação em rede, permitindo que os desenvolvedores foquem na lógica de negócios.

Conclusão:

Este projeto demonstra como o RPC pode ser aplicado de forma eficaz em um cenário real como um sistema de gestão de livreria online. Ele possibilita uma comunicação sem problemas entre aplicativos clientes e componentes do lado do servidor, facilitando interações suaves dos usuários e o gerenciamento eficiente das operações da livreria.

Não utilize base de dados no servidor, carregar catálogo em memória.