**Giao thức grpc**

gRPC (gRPC Remote Procedure Call) là một giao thức mã nguồn mở do Google phát triển, được sử dụng để xây dựng các dịch vụ phân tán. Nó cho phép các ứng dụng giao tiếp với nhau thông qua các cuộc gọi hàm từ xa (RPC - Remote Procedure Call). gRPC sử dụng giao thức HTTP/2, Protobuf (Protocol Buffers) để tuần tự hóa dữ liệu, và cung cấp các tính năng như xác thực, cân bằng tải, theo dõi, và nhiều tính năng khác.

- HTTP/2 là phiên bản thứ hai của giao thức HTTP, được thiết kế để cải thiện hiệu suất của các ứng dụng web bằng cách tối ưu hóa tốc độ truyền tải và giảm độ trễ. Cho phép nhiều yêu cầu và phản hồi trên cùng một kết nối TCP, giúp tăng hiệu suất và giảm độ trễ. Cho phép server gửi trước các tài nguyên đến client mà không cần client yêu cầu.

- Protobuf (Protocol Buffers) là một ngôn ngữ mô tả giao diện (IDL) và công cụ tuần tự hóa dữ liệu do Google phát triển, thường được sử dụng để trao đổi dữ liệu giữa các ứng dụng với hiệu suất cao. Tuần tự hóa và giải tuần tự hóa nhanh hơn so với JSON và XML. Sử dụng các file .proto để định nghĩa cấu trúc dữ liệu.

Các bước thực hiện

- File .proto giúp định nghĩa rõ ràng các dịch vụ và cấu trúc dữ liệu cần thiết cho gRPC. Nó đóng vai trò là hợp đồng giữa client và server, đảm bảo rằng cả hai bên hiểu rõ cách thức giao tiếp và định dạng dữ liệu.

**Các phương thức trong grpc**

- Unary RPC: Một phương thức RPC truyền một yêu cầu từ client đến server và nhận một phản hồi duy nhất. Đây tương tự như một yêu cầu POST trong RESTful API

- Server Streaming RPC: Client gửi một yêu cầu đến server và nhận một luồng các phản hồi từ server. Tương tự như một yêu cầu GET trong RESTful API

- Client Streaming RPC: Client gửi một luồng yêu cầu đến server và nhận một phản hồi duy nhất từ server. Tương tự như POST hoặc PUT trong RESTful API

- Bidirectional Streaming RPC: Cả client và server đều gửi và nhận luồng dữ liệu đồng thời