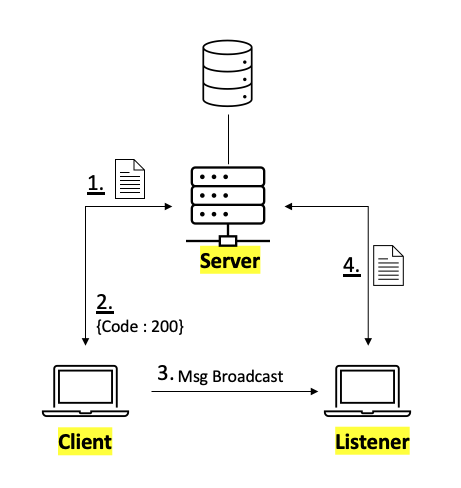
**-프로젝트 설명**

Golang, gRPC 통신, 동시성 프로그래밍을 적용하여 데이터 저장 및 알림 기능 구현을 연습하기 위한 프로젝트 입니다.

**-프로젝트 구조 및 동작 순서**



**- Server :** gRPC 요청을 처리하여 Client로부터 요청된 데이터를 저장하는 역할을 수행

**- Client :** Server로 저장할 데이터를 전송하여 데이터 저장 후 저장된 데이터의 Index를 Listener로 전송

**- Listener :** Client가 전송한 Index를 받은 후 Server로 데이터를 요청

**위 3가지로 구성된 모듈이 아래 4가지 과정을 순서대로 수행해야 합니다.**

1. Client는 Server로 gRPC 통신을 통해 데이터 저장 요청을 전송합니다.   
   (해당 과정을 구현할 때 Client 상에서 gRPC Call은 Go Thread를 통해 실행)
2. Server는 Client가 전송한 요청을 수행하여 데이터베이스에 데이터를 저장하고 응답 코드(성공 시 200, 실패 시 500 반환)와 데이터의 Index를 반환합니다.
3. Client는 Thread에서 실행된 gRPC Call 요청이 끝난 후 해당 응답을 Golang의 채널 기능을 통해 받은 후 Listener로 Index를 전송합니다. (Client-Listener 간 통신은 단방향이 아닌 양방향 통신을 사용합니다.)
4. Listener는 Index를 서버로 보내 (1)에서 저장된 데이터를 반환받습니다.

**-프로젝트 폴더 구조**

텍스트이(가) 표시된 사진

자동 생성된 설명

프로젝트 구현 예시

* GRPC 서버 동작을 위해 Service proto 파일을 작성 후 빌드하여 서버 및 클라이언트에서 사용하면 됩니다.
* Server, Client, Listener 각각 다른 Main 파일을 만들어 실행하면 됩니다.

텍스트, 장치, 시계, 측정기이(가) 표시된 사진

자동 생성된 설명

프로젝트 모듈 설정

텍스트, 실내, 스크린샷이(가) 표시된 사진

자동 생성된 설명

gRPC 예시 : 서비스 목록 (Server에서 동작)

**- 추가 내용**

해당 프로젝트를 구현하며 추후 다른 프로젝트에서도 사용할 수 있도록 아래 사항들을 정리(소스코드, 참고한 블로그 등)하세요

1. Golang 설치, 환경변수 설정 방법
2. Go module 생성 방법
3. Nosql CRUD
4. gRPC Server, Client 실행
5. 양방향 통신 방법