gRPC support for API Dash

There are 2 approaches to this which are mentioned below

1. Create an embedded golang service that communicates with the Flutter Dart client and makes the grpc calls based on the parameters provided.

   Advantages -
   - We can support server reflections alongside standard proto-file parsing

   Challenges -
   - Bit complex in implementation
   - Need some rnd around implementation

   HLD For Golang Service -
   - Function MakeCall - Gets the arguments and initiates a grpc call according to params, returns the response and error if there
   - Function GetServerReflections - The client calls this to fetch the list of services running which support server reflections. It will return the reflection data

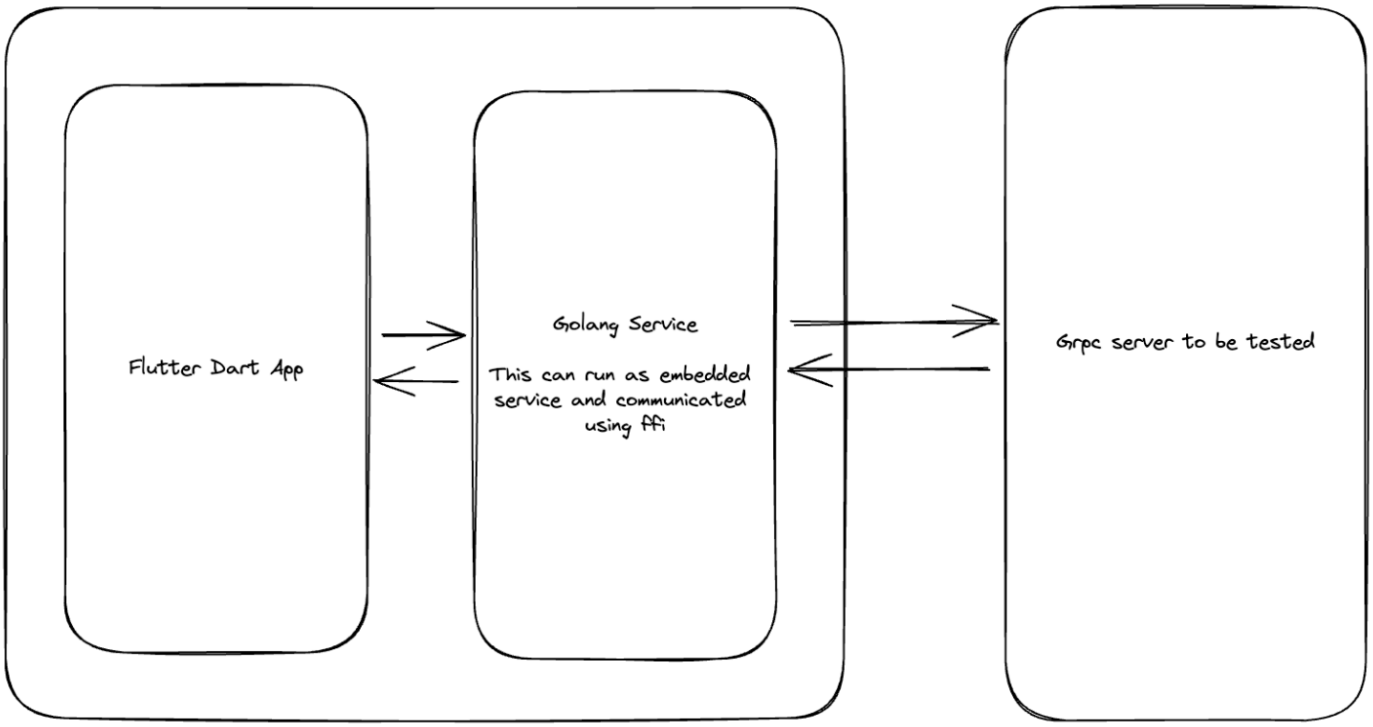
2. Use grpc:dart and communicate from the flutter dart client directly to the service to be tested

   Advantages -
   - Easier and simpler implementation

   Disadvantages -
   - We cannot support server reflections as grpc:dart does not support that

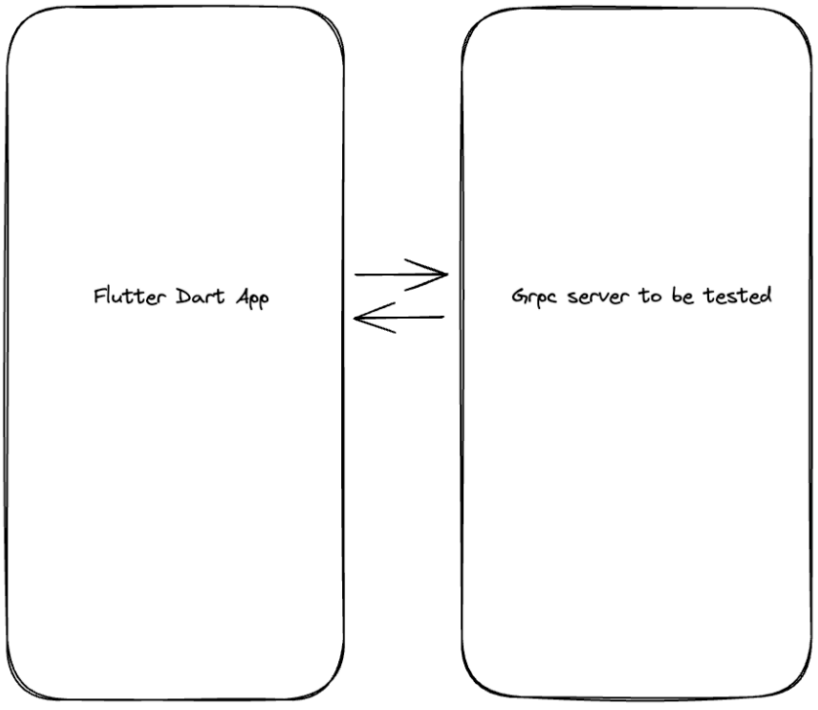
References -
https://github.com/fullstorydev/grpcurl
https://github.com/fullstorydev/grpcui

First Approach



Benefits of this approach is we will be able to support server reflections in grpc

Second Approach



This is simpler to implement as we can use grpc:dart package and communicate directly
The disadvantage is that we won't be able to support server reflections as grpc:dart doesn't have that