CSCI 320: Networking and Distributed Computing
Project 02 Timing

Riley Kirkpatrick

April 12th, 2020

After 9 runs of gRPC timing, I selected the run with the median total time to run. The following times are from that run and are divided into how long each function took to time (all times in seconds).

GetProductsByID: 0.15118874400000015

GetProductsByName: 0.1221241700000002

GetProductsByManufacturer: 0.11720750000000013

GetOrdersByID: 0.07756279999999993

GetOrdersByStatus: 0.15648306599999984

GetProductsInStock: 0.10037608600000025

UpdateProducts: 0.38135878300000003

UpdateOrders: 0.20439743599999982

AddProducts: 0.13557887199999996

CreateOrders: 0.21843332399999937

Total: 1.6647107809999997

All functions interacting with products use 500 products, whether that means the function retrieves, adds, or updates them. Similarly, all functions interacting with orders use 50 orders, whether that means the function retrieves, adds, or updates them.

The choice of 500 products comes from the fact that gRPC has a 4 MB message size limit and using 750 or 1000 (or more) products exceeds this byte limit. In the future I could modify my program to use bidirectional streaming to overcome this limitation, with the caveat of increased complexity. Another way to handle large amounts of products and orders is to call the gRPC functions multiple times and only pass up to 500 products/orders at a time.

TODO: Time by calling gRPC functions multiple times so that total time is around 15+ seconds.