The project1 requires us to design a server to operate key-value pairs. And the server needs to support both TCP and UDP communication. In order to provide two ways of communication, I run the servers in two seperate ports, and let the client have the options of TCP/UDP communication. In my opinion, the split of servers is like simple distributed systems, the two servers provide different services for the client. And currently the two servers run in the same machine, but if the service gets heavier, the servers might need to run in two separate machines, or even multiple machines.

This project also requires us to focus on writing robust service, like the timeout setting, malformed packet handling, program log and so on. I added input error detection on both client and server sides and added logs for important processes. And since there are some services being commonly used, like log service and map service, I create classes for the two services to reduce code redundancy. Also since the two servers need to share common data for key-value pairs, the independent map class stores the data from the two servers together.

To make the client and servers provide continuous services, I use a while loop to let the services keep running until exceptions.