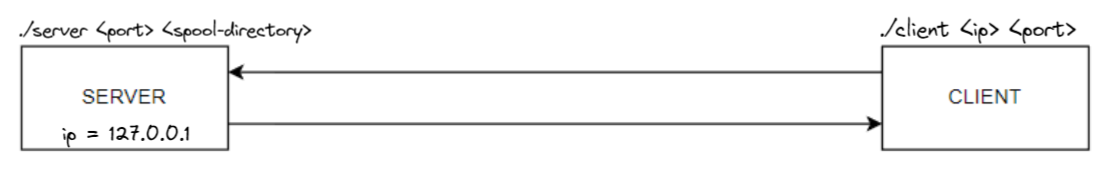
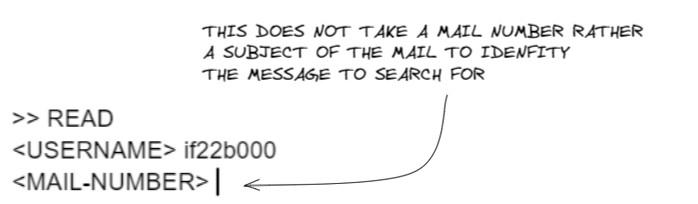
## Server Architecture

Our server can only handle one client at a time. The server is able to handle the following commands SEND, READ, DEL, LIST, QUIT. If a client send another command it will be ignored.

For the current implementation we decided that READ, DEL and LIST will use the message subject to identify the message a user wants to access. We have not implemented ids or basically our ids are the subjects :)

Command Usage example:



## Used technologies

We used following tools

* git (version control)
* VS Code LiveSharing (Pair programming)
* VS Code (as Editor)

## Development Strategy

Most of the time spent developing was via Pair Programming. It made us more efficient than when solo developing which also sped up development time. Since we were able to find the root cause of errors significantly faster.

## Adaptions

Currently our codebase is a mixture of C and C++ code for the final version we are going to rewrite the C code to C++ so we have a uniform code base.

In addition, we have stil many edge cases and constraints that have to be check and handled this will also be included in upcoming versions.