What works:

- I completed all request messages
    - Each request will return a dictionary that states the message number, and what the peer was requesting for
- Error checking is implements on most parts of the program
- When a user is requesting anything, there will be a list provided to choose from.
- My program can support multiple clients and clients can directly contact each other
- My program supports parallel downloading
- All dictionaries are updated with the appropriate data when a peer receives a chunk
- All requested chunks are written in order

What doesn't work:

- I did not implement an integrity check
- When a peer finishes downloading all chunks individually, there is no way to confirm in the system that it is the whole file.
- There is no view on active downloads
- When a peer closes out, the information of what files and chunks the peer has, still remains in the dictionary stating that the peer is still present.
- Does not take into account if a peer requests the same chunk twice.
- If there are no files shared at all in the server and you go to request file locations, it will error out. There is no error checking if no files are shared