Basic Network Design:

- RabbitMQ handles all networking/TCP/IP communications
    - Our server reads and writes to message queue(s)
    - Clients read and write to message queue(s)
        - Client-server communication queues are named after User UIDs
    - IMPORTANT: Client and our Server only need to connect to the RabbitMQ server
        - There is no need to track IPs; that is all handled by the RabbitMQ server
- RabbitMQ does NOT provide a method to parse information
    - Data is just a stream that we can get a string off of
    - We will use JSON
        - RabbitMQ “message bodies” will store JSON strings
        - The RabbitMQ body can be stripped out when a message is received and then a JSON parser (Python json package, jsonRPC) can convert the string to application-specific information

- Queue Diagrams
  All Clients → Default Queue → Server
  Server → Bootstrap Queue or Exchange on Default Queue (?) → Client gets UID and now can connect to a user-specific queue that it’s listening on for server messages
  Server → UID Queue (per Client) → Client