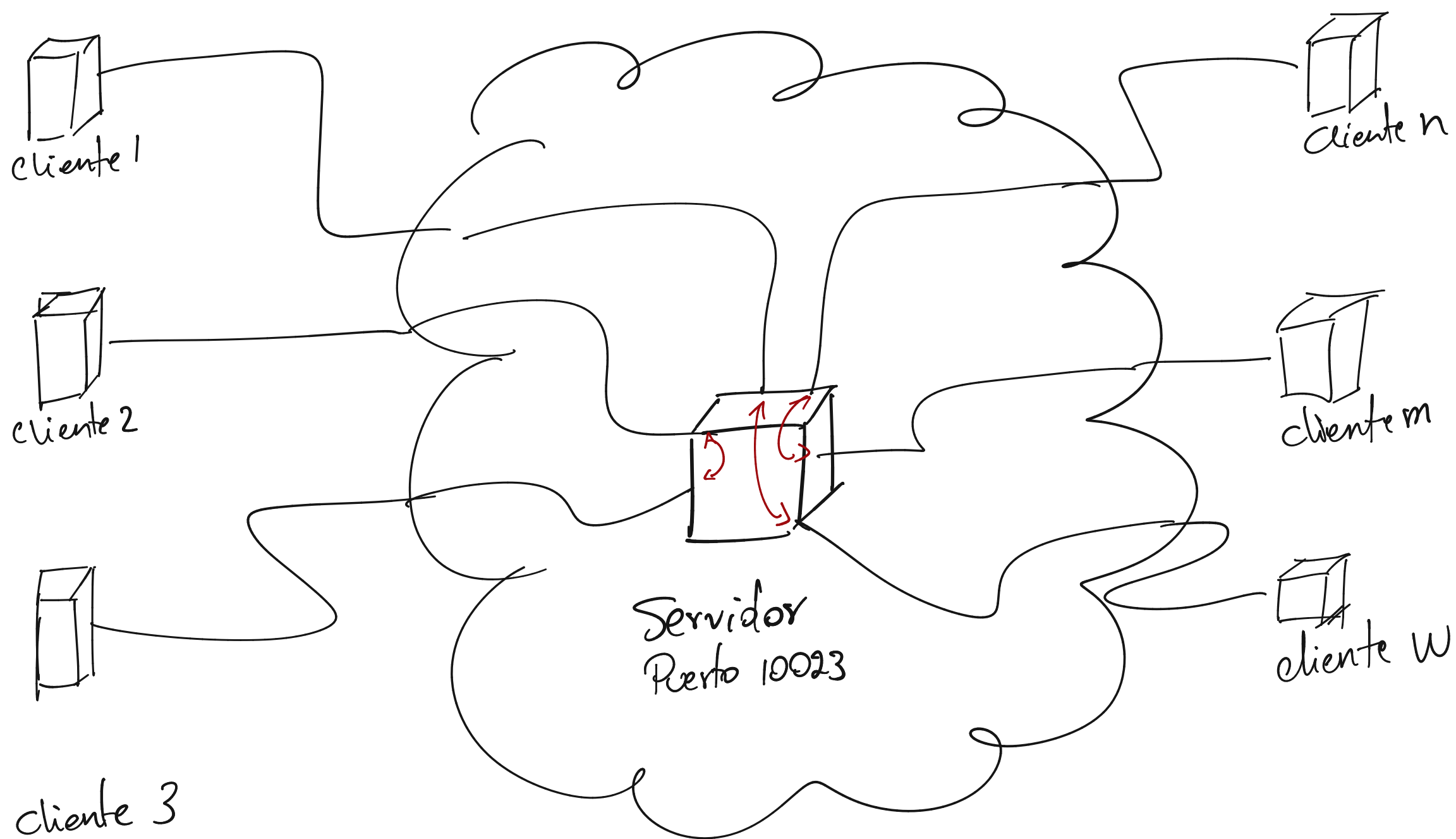


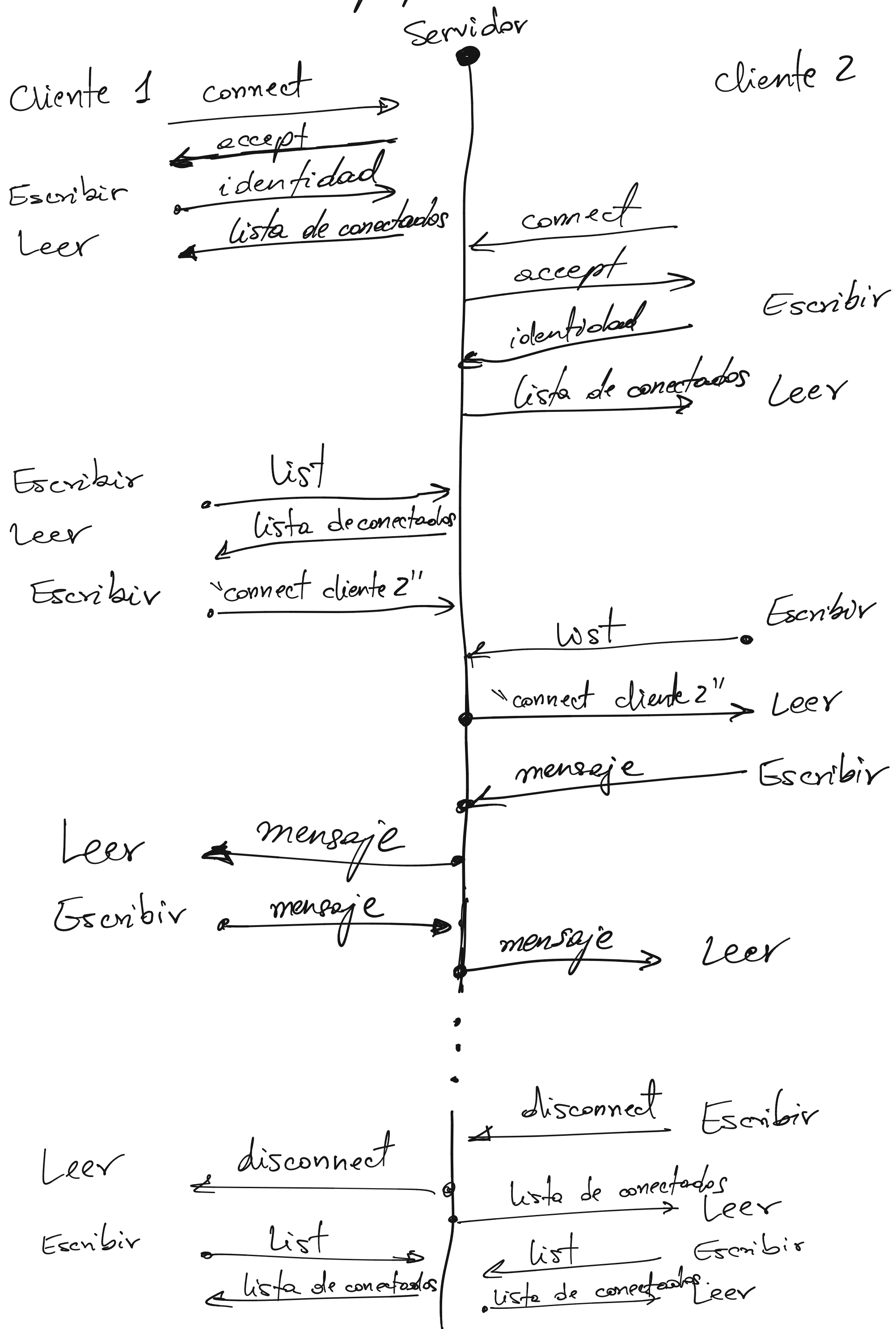
Tarea 3

Chat Sincrónico Centralizado



- ① Existe un solo servidor.
- ② Servidor hace los veces de proxy para comunicar dos clientes.
- ③ Cada cliente compartirá su identidad (Nombre)
- ④ Servidor tendrá una lista de los clientes conectados.
- ⑤ Usaremos comandos en inglés:
 - a) "list" para pedir la lista de personas / clientes conectados. Este comando se ejecutará cada 10 segundos mientras un cliente no tome otra acción.
 - b) "connect" para pedir conexión con un usuario disponible.
 - c) "disconnect" para terminar una conexión con un cliente o con el servidor.
- ⑥ La comunicación será síncrona:
El servidor siempre esperará un comando y luego escribirá en el socket.
El cliente siempre escribirá en el socket primero y luego esperará un comando.

⑦ Cuando se comuniquen dos procesos "cliente" en el servidor el cliente que inicia el chat deberá esperar primero. Para esto el servidor preparará el socket del cliente.



- ⑧ Tanto cliente como servidor serán escritos en Python, y estarán basados en el cliente/servidor escritos en clase.
- ⑨ La función "receive" usará un buffer de 1024 bytes.
- ⑩ El servidor correrá en la nube.
- ⑪ Será configurado para recibir conexiones desde el Internet.
- ⑫ Si todos seguimos las reglas de juego del protocolo de la capa de aplicación, este chat debe ser compatible clientes y servidores entre grupos de la clase.