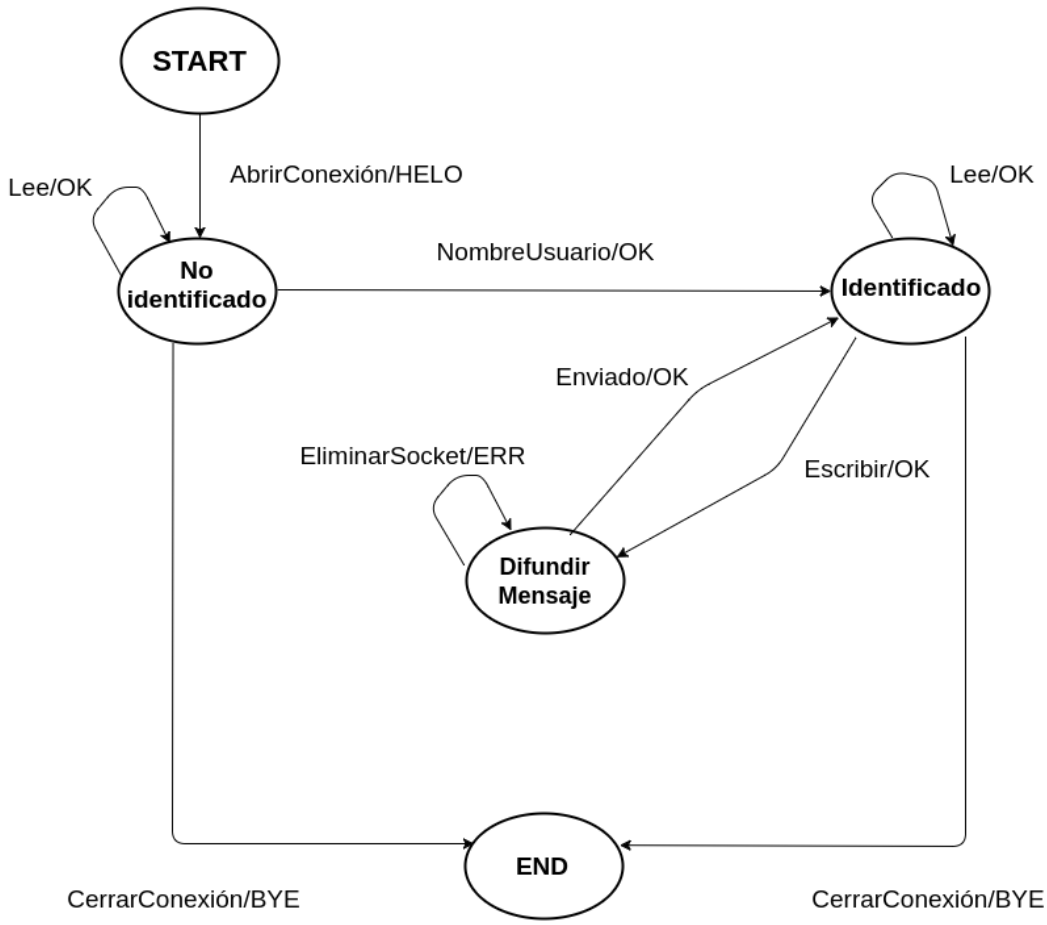
Aplicación cliente-servidor: Chat multicliente.

1. **Descripción de la aplicación.**

Nuestra aplicación se trata de un chat para intercambiar mensajes en texto plano entre dos clientes. Dicho intercambio es posible gracias al servidor, que actúa como intermediario tratando las peticiones de cada cliente y respondiendo de forma adecuada a cada una de ellas.
Por tanto, la aplicación permite la comunicación entre dos o más clientes mediante texto, y aunque hayamos hecho la aplicación en local, bastaría con cambiar las IPs a la que dirige el servidor los mensajes de cada cliente.
La aplicación se sirve del protocolo TCP, para establecer la conexión entre el servidor y los clientes. Además, dicho protocolo controla todo lo relacionado al tratamiento de los mensajes, por ejemplo comprobar que los mensajes han llegado y el control de errores.

2. **Diagrama de estados del servidor.**

El diagrama de estados por los que pasa el servidor es el siguiente:



3. Mensajes gestionados en el servidor.

Los mensajes que trata el cliente se muestran en la siguiente tabla:

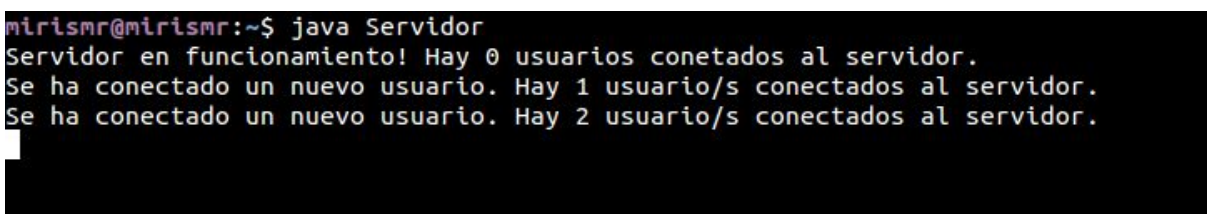
Código	Cuerpo	Descripción
100	ABRIRCONEXION + direccion_servidor + puerto	Abre la conexión cliente-servidor e inicializa el socket.
200	NOMBREUSUARIO + nombre	Asigna el nombre de usuario al cliente.
300	LEE	Lee los mensajes pendientes del socket asociado al servidor.
301	ESCRIBIR + mensaje	Escribe el mensaje en el socket asociado al servidor.
400	CERRARCONEXION	Cierra la conexión cliente-servidor.

Los mensajes que trata el servidor se muestran en la siguiente tabla:

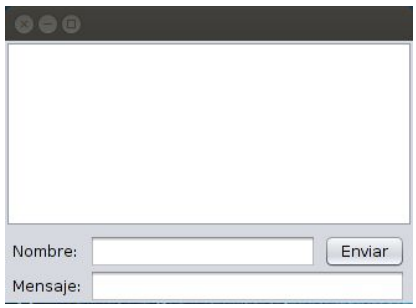
Código	Cuerpo	Descripción
302	DIFUNDIRMENSAJES + mensajes	Lee los mensajes pendientes de enviar y los envía a todos los clientes conectados.
401	ELIMINARSOCKET	Elimina el socket de un usuario tras su desconexión.

4. Evaluación de la aplicación.

Como la aplicación se basa en el modelo cliente-servidor, es evidente que lo primero que debemos hacer para ponerla en marcha será ejecutar el servidor. No se ha diseñado interfaz gráfica (al menos en el prototipo), pues no parece ser realmente interesante, ya que solo se encargará de transmitir los mensajes de un cliente al resto. El servidor simplemente mostrará el número de clientes conectados en ese momento. Podemos verlo en la siguiente figura:



Para los clientes, se ha diseñado una interfaz gráfica simple, donde podemos introducir el nombre de usuario de cada cliente, así a la hora de leer los mensajes de cada uno de los clientes se puede identificar qué cliente mandó cierto mensaje. Como es lógico, en dicha interfaz se reserva un campo para introducir el texto que cada cliente quiera mandar, así como un cuadro de texto donde se verán reflejados los mensajes que cada cliente ha enviado. Se muestra la interfaz en la siguiente figura:



Un ejemplo del uso del chat se puede observar en la siguiente imagen: