

LABORATORIO 4

Crear un cliente de mensajería por sockets TCP. El cliente debe permitir enviar y recibir mensajes en todo momento.

El diseño de la interfaz gráfica es libre, se muestra a continuación una a modo de ejemplo:

The screenshot shows a graphical user interface for a TCP messaging client. It features a light gray background. On the left side, there are two input fields labeled 'host' and 'port', followed by a blue button labeled 'Add Contact'. Below these is a list box labeled 'Lista de contactos'. In the center, there is a large text area labeled 'Text' with a vertical scrollbar. Below the text area, there is a green label 'Envíe un mensaje ...' and three buttons: 'Send', 'Send File', and 'Clear'. At the bottom, there is a section labeled 'Incomming messages' with a large empty rectangular box.

El cliente deberá tener las siguientes funcionalidades:

- Enviar un mensaje de texto indicando el host / puerto del destinatario
- En caso de no poder enviar el mensaje indicarlo por pantalla
- Recibir en forma paralela mensaje e informar los mismos por pantalla indicando el host del remitente del mensaje

Para lograr esto vamos a necesitar como mínimo las siguientes clases:

EnviarDeMensajes

Esta clase va a ser un thread o runnable y se instanciará y arrancará al momento de hacer click en el botón “enviar”.

Su responsabilidad será:

- leer los datos de pantalla (host destino, puerto destino y mensaje)
- abrir una conexión por socket tcp al destino
- enviar el mensaje
- cerrar el socket

Muy similar a lo que hacía nuestro Cliente TCP

ReceptorDeMensajes

Esta clase va a ser un thread o runnable y se instanciará y arrancará por única vez al iniciar nuestra aplicación y correrá mientras la aplicación siga levantada.

Su responsabilidad será:

- abrir un ServerSocket en el puerto deseado para la recepción de mensajes
- por cada conexión recibida pasarle socket a una instancia de la clase ManejadorDeMensajes y arrancarla.
- debe funcionar en bucle infinito
- Muy similar a lo que hacía nuestro Server TCP

ManejadorDeMensajes

Esta clase va a ser un thread o runnable y se será instanciada y arrancada por el ReceptorDeMensajes cada vez que reciba una nueva conexión.

Su responsabilidad será:

- leer el mensaje enviado
- mostrar el mensaje en la interfaz gráfica
- cerrar el socket

Para lograr el envío de archivos similar estructura de clases en necesaria.