Proyecto chat cifrado

Requerimientos funcionales

1. **RF1:** El sistema debe permitir que un usuario actúe como servidor y otro como cliente
2. **RF2:** El sistema debe permitir la conexión entre dos computadoras por dirección Ipy puerto
3. **RF3:** El sistema debe realizar un intercambio de claves mediante el algoritmo Diffie-Hellman.
4. **RF4:** El sistema debe derivar una clave de 256 bits a partir de la clave compartida generada
5. **RF5:** El sistema debe usar AES-256 en modo CBC o GCM para cifrar los mensajes
6. **RF6:** El sistema debe mostrar un mensaje de conexión exitosa cuando el canal esté seguro
7. **RF7:** El sistema debe permitir al usuario enviar mensajes de texto una vez establecida la conexión segura
8. **RF8:** El sistema debe permitir recibir y descifrar mensajes desde el otro extremo
9. **RF9:** El sistema debe cifrar cada mensaje antes de enviarlo por la red
10. **RF10:** El sistema debe permitir descifrar cada mensaje recibido utilizando la clave AES
11. **RF11:** El sistema debe permitir finalizar la conexión de manera segura
12. **RF12:** El sistema debe registrar errores de conexión o cifrado para su diagnóstico
13. **RF13:** El sistema debe validar que no se puedan enviar mensajes antes del establecimiento de la clave
14. **RF14:** El sistema debe permitir limpiar el área de chat local si el usuario lo desea
15. **RF15:** El sistema debe permitir configurar el puerto de conexión antes de iniciar
16. **RF16:** El sistema debe validar el tamaño y formato de los mensajes antes de enviarlos
17. **RF17:** El sistema debe avisar al usuario cuando la conexión remota se pierde
18. **RF18:** El sistema debe cerrar el socket y liberar recursos al terminar la sesión
19. **RF19:** El sistema debe tener una interfaz básica en consola para la interacción con el usuario
20. **RF20:** El sistema debe evitar el reenvío de mensajes ya enviados (no retransmitir automáticamente)