



PROYECTO UNIDAD 1:

Chat Offline con Gestión de Conversaciones en XML

Introducción

El proyecto consiste en la implementación de un **CHAT OFFLINE** donde los mensajes se almacenan en un archivo XML. El sistema tendrá funcionalidades como el registro de usuarios, la capacidad de enviar y recibir mensajes, y la posibilidad de generar resúmenes o informes de las conversaciones utilizando Streams. El archivo XML servirá como la base de datos de las conversaciones, permitiendo guardar y consultar los mensajes entre los usuarios.

Funcionalidades:

1. **Registro de usuarios:** Los usuarios del chat deben registrarse. El sistema almacenará los usuarios registrados en un archivo XML separado.
2. **Envío de mensajes:** Los usuarios registrados pueden enviar mensajes entre sí. Debes gestionar esta opción con XML. No se deben poder borrar los mensajes
3. **Visualización de conversaciones:** Cuando un usuario entra en el chat y elige un remitente puede ver todas las conversaciones antiguas.
4. **Generar resumen de conversación (usando Streams):** Se podrá generar un informe de la conversación entre dos usuarios, utilizando Streams para contar el número de mensajes, las palabras más comunes, o generar estadísticas simples sobre la conversación. En este punto puedes ampliar todo lo que quieras.
5. **Exportar conversación a texto plano:** Opción para exportar las conversaciones a un archivo de texto para archivarlas o visualizarlas en un formato más simple.
6. **Uso de archivos adjuntos:** Permitir adjuntar archivos (imágenes, PDF, etc.) a los mensajes:
 - Guardar los adjuntos en una carpeta media/ usando Streams binarios (InputStream/OutputStream).
 - Asociar en el XML del mensaje una referencia al adjunto.
7. Implementar una opción para empaquetar conversaciones + adjuntos en un ZIP usando ZipOutputStream. (opcional)

Requisitos:

1. **Lectura y escritura en XML:** Implementa métodos para leer y escribir tanto en el archivo de usuarios como en los de conversaciones utilizando **JAXB**.
2. **Registro de usuarios:** Debe haber un método para añadir nuevos usuarios al archivo XML de usuarios.
3. **Envío de mensajes:** Al enviar un mensaje, este se debe agregar a la conversación correspondiente en el archivo XML de mensajes.
4. **Consulta de conversaciones:** Se deben poder recuperar todas las conversaciones entre dos usuarios, leyendo y filtrando los mensajes desde el archivo XML.
5. **Streams para análisis de conversación:** Usa Streams para realizar operaciones como contar el número total de mensajes en una conversación, o extraer información adicional como las palabras más repetidas, la cantidad de mensajes por usuario, etc.
6. **Exportar conversación:** Implementa un método que exporte una conversación entre dos usuarios a un archivo de texto o CSV, utilizando Streams para procesar los datos.
7. **Interfaz:** Se puede programar con interfaz gráfica (JavaFX)
8. **Uso de archivos adjuntos:**
 1. Guardar los archivos adjuntos en una carpeta media/ usando **InputStream/OutputStream**.
 2. Asociar en el XML del mensaje una referencia al adjunto (nombre, ruta, tamaño, tipo).
 3. Validar tamaño y extensión de archivos antes de guardarlos.
 4. Permitir abrir o exportar los adjuntos desde el chat.
 5. Si un adjunto falta, debe marcarse en la conversación pero no borrarse el mensaje.

Entrega:

1. PDF con: modelo, detalles a destacar y enlace a git.
2. Será necesario realizar una demo con presentación para su corrección.
3. Se permite realizar trabajo en grupos de 2, o individual.