

UNIDAD DE APRENDIZAJE : Aplicaciones para comunicaciones en red	
UNIDAD TEMÁTICA I: Sockets de flujo	
No. Y Título de la práctica:	Tiempo de realización: 3 horas
Práctica no. 1 Servicio de transferencia de archivos	
Objetivo de la práctica: El estudiante implementará una aplicación para el envío de múltiples archivos a través de la red haciendo uso de sockets de flujo	
Situación problemática: En muchas organizaciones se requiere transferir grandes volúmenes de archivos a través de la red (archivos de audio, video, fotografías, archivos ejecutables, etc.) de una manera confiable, es decir, garantizar la entrega de los datos, que estos lleguen en orden y sin duplicados.	
¿Qué tipo de socket permite garantizar la entrega de datos cubriendo todas las necesidades anteriormente descritas?	
Competencia específica: Desarrolla aplicaciones en red, con base en el modelo cliente-servidor y utilizando únicamente la interfaz de sockets de flujo bloqueantes, para el envío de datos.	
Competencias genéricas: <ul style="list-style-type: none">• Aplica los conocimientos en la práctica• Demuestra habilidad para trabajar en equipo• Demuestra capacidad de investigación• Desarrolla aplicaciones en red con base en la tecnología más adecuada	Elementos de competencia: <ul style="list-style-type: none">• Programa aplicaciones en red con base en el modelo Cliente-Servidor y la interfaz de aplicaciones de sockets de flujo• Analiza los servicios definidos en la capa de transporte• Emplea el modelo Cliente-Servidor para construir aplicaciones en red• Programa aplicaciones Cliente-Servidor utilizando sockets de flujo bloqueantes
Criterios de evaluación: Las prácticas 1, 2 y 3 aportarán el 30% de la unidad temática I	

Rúbrica (analítica) para la U.A. Aplicaciones para comunicaciones en red.

Producto: Servicio de transferencia de archivos

Valoración: Novato (0-150pts), Intermedio (151-300 pts), Avanzado (301-450 pts), Experto (451-600 pts)

ASPECTOS A EVALUAR	Excelente (100pts)	Cumplió bien (75pts)	Cumplió (50pts)	No satisfactorio(25pts)
Análisis	Entiende el problema a resolver, haciendo uso de los elementos de programación en red precisos	Utiliza los elementos de programación en red necesarios	Utiliza algunos de los elementos de programación en red	No tiene idea de cómo resolver el problema, ni que tipo de comunicación necesita implementar
Diseño	Define una arquitectura de comunicación en base al tipo de comunicación a utilizar	Define una arquitectura de comunicación que puede o no ser la mejor opción para el tipo de comunicación a utilizar	Define una arquitectura de comunicación que no se ajusta al tipo de comunicación	No define ninguna arquitectura
Implementación	Utiliza los elementos solicitados(sockets de flujo, flujos orientados a byte) y desarrolla la aplicación de acuerdo a los requerimientos	Utiliza alguno(s) de los elementos solicitados y desarrolla la aplicación de acuerdo a los requerimientos	Utiliza alguno(s) de los elementos solicitados, pero no desarrolla la aplicación de acuerdo a los requerimientos	No utiliza ninguno de los elementos y no desarrolla la aplicación en base a los requerimientos
Conocimientos	Muestra dominio de los elementos (sockets de flujo bloqueantes, flujos orientados a byte, archivos)	Muestra dominio de los elementos, así como de programación	Muestra dominio de algunos de los elementos	No muestra dominio de los elementos, ni de programación, ni de diseño de interfaces

Presentación	Dominio de los temas, explicación de algoritmos usados utilizados, caso de prueba bien diseñado	Explicación de algoritmos utilizados	Supo darse a entender al explicar algunos de los algoritmos	No supo darse a entender, se confundió
Trabajo colaborativo	Desde el desarrollo del proyecto se denotó el trabajo en grupo, la empatía, buena distribución de tareas, colaboración de todos. Todos saben sobre el proyecto en su totalidad.	Desde el desarrollo del proyecto se denotó el trabajo en grupo, existen algunos roces de opiniones y malentendidos , colaboración de todos. Todos saben sobre el tema en su totalidad.	Desde el desarrollo del proyecto se denotó el trabajo en grupo, la empatía, buena distribución de tareas, colaboración de todos.	Desorganización, falta de interés, el trabajo no se repartió equitativamente, huecos de ignorancia en el desarrollo del proyecto

Introducción

El envío de archivos a través de la red es una característica importante para la gran mayoría de las aplicaciones que hoy día se utilizan (blogs, redes sociales, mensajería instantánea, Declaración de impuestos, educación en línea, etc.), sin embargo, no todas las aplicaciones disponibles permiten el envío de archivos de gran tamaño (p.e. El correo electrónico no permite enviar archivos de más de 10 o 20 MB). Esto hace necesario el desarrollo de aplicaciones que permitan transferir archivos sin importar el tamaño de éstos.

Recursos y/o materiales

- | | |
|---|--|
| <ul style="list-style-type: none">• Manual de prácticas de laboratorio de Aplicaciones para Comunicaciones en Red• Plumones• Bibliografía | <ul style="list-style-type: none">• Internet• Computadora• IDE de desarrollo• Apuntes |
|---|--|

Instrucciones

En esta práctica debes implementar un servicio de transferencia de archivos para que el cliente de la aplicación pueda enviar/recibir uno o más archivos de cualquier tamaño/tipo hacia/desde el servidor. Además, deberás dar al usuario, desde la aplicación cliente, la posibilidad de modificar parámetros de la comunicación mediante el uso de opciones de socket (Habilitar/deshabilitar el algoritmo de Nagle, modificar el tamaño de los buffers de escritura/lectura, Modificar temporizador de lectura, etc.) para modificar el desempeño de la aplicación

Desarrollo de la práctica

A partir de los programas CEnvia y SRecibe que te serán proporcionados por el profesor deberás realizar las siguientes modificaciones:

- La aplicación cliente deberá mostrar al usuario un menú que permita realizar las siguientes acciones: subir archivos/carpetas a la carpeta remota en el servidor, descargar archivos/carpetas hacia la carpeta local del usuario en el cliente, listar el contenido ya sea de la carpeta local (o la carpeta remota) indicando para cada entrada de la carpeta si se trata de un archivo o carpeta. También se dará la posibilidad de eliminar archivos/carpetas de manera local/remota.
- El programa CEnvia implementa un cliente de flujo bloqueante que realiza el envío de un archivo elegido previamente a través de una caja de diálogo (JFileChooser) y éste es enviado utilizando flujos orientados a byte. Deberás modificar el código para que en lugar de enviar un archivo, este programa permita enviar cualquier cantidad de archivos (secuencialmente) o carpetas. Para esto, la aplicación hará uso de dos sockets de flujo (uno que siempre estará abierto para el envío de metainformación, el segundo solo se abrirá cada que se transmita el contenido de un archivo y posteriormente será cerrado).
- El programa SRecibe implementa un servidor de flujo bloqueante que recibe un archivo por cada conexión

aceptada y éste es recibido usando flujos orientados a byte. Deberás modificar el código para que en lugar de recibir un solo archivo, éste sea capaz de recibir/enviar uno o más archivos o carpetas. Cada archivo se enviará de manera individual.

- Realiza pruebas intentando enviar distintos tipos de archivo (imágenes, texto, ejecutables), así mismo intenta enviar archivos de distintos tamaños (menos de 100KB, más de 100KB y menos de 10MB, más de 10MB y menos de 200MB, más de 200MB y hasta 2GB).

Cierre de la práctica

Preguntas:

1. ¿Qué tipo de archivos se enviaron más rápido?
2. ¿Cuál fue el número máximo de archivos que fue posible enviar a la vez?
3. ¿Cuál fue el tamaño de archivo más grande que se pudo transferir? ¿por qué?
4. ¿Qué es el orden de red?
5. ¿Por qué razón es importante utilizar el orden de red al enviar los datos a través de un socket?
6. Si deseáramos enviar archivos de tamaño muy grande, ¿qué cambios sería necesario hacer con respecto a los tipos de datos usados para medir el tamaño de los archivos, así como para leer bloques de datos del archivo?