

Introducción a las Redes de Computadoras

(IRC)² Client

Introducción a las Redes de Computadoras – Internet Relay Chat (like) Client

2do. TRABAJO OBLIGATORIO

Versión 0.2 - 27/03/10

Se desea implementar una aplicación simple de comunicación sobre Internet que permita el intercambio de información (mensajería) entre múltiples usuarios.

Introducción

La entrega debe hacerse únicamente mediante el formulario destinado a tales fines en <http://www.fing.edu.uy/inco/cursos/redescomp/> en la etiqueta Entrega. Se podrán hacer varias entregas, pero la única válida será la última dentro del plazo asignado.

Se debe entregar un solo archivo 'ob2.tar.gz' que contenga los ítems descritos en el apartado "Entrega". Dicho archivo deberá ser generado utilizando la herramienta GNU TAR y compresión gzip. Otros formatos (bz2, rar, zip, cab, jar, etc.) serán rechazados.

Los trabajos se recibirán hasta el día 18 de abril de 2010 a las 23:30 horas, sin excepciones.

La corrección se hará en la máquina virtual del curso en el salón 202, por lo que debe probarse su funcionamiento en la sala previo a la entrega.

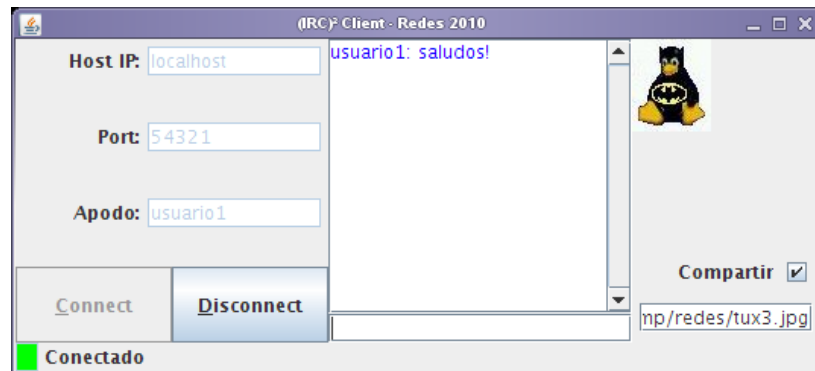
<p>Está prohibido, bajo pena de sanciones, utilizar o copiar código de otros grupos, así como publicar código en cualquier medio</p>

Descripción del problema

Se desea implementar una aplicación (cliente y servidor) de discusión en línea para grupos sobre Internet. La misma utilizará un protocolo de transporte confiable, orientado a conexión (TCP) y permitirá el intercambio de mensajes entre todos los clientes conectados simultáneamente.

Descripción del cliente

El cliente será programado en Java, partiendo preferentemente de la GUI suministrada, de acuerdo a la siguiente figura:



Previo a iniciar la conexión el usuario podrá seleccionar la dirección del servidor, el puerto en el que atenderá, así como, su apodo a lo largo de la sesión. Luego que el cliente se conecte al servidor, podrá mandar mensajes y recibir los mensajes de otros clientes conectados. Los mensajes mostrados llevarán como prefijo el apodo de quién lo envía, que no podrá ser modificable durante la conexión.

Es responsabilidad del usuario suministrar la dirección IP (o nombre de host) y puerto en el que atiende el servicio. El puerto predeterminado será el 54321.

Sobre la derecha de la GUI hay un panel que muestra la imagen asociada al último usuario que envió un mensaje (si éste comparte alguna imagen) o nada, si el usuario no quiere compartir imagen alguna. Cabe mencionar que es responsabilidad del usuario suministrar imágenes válidas y de tamaños coherentes, no se espera que la GUI escale las imágenes de forma alguna.

El servidor de imágenes implementado en cada aplicación cliente deberá ser multithreaded y aceptar conexiones concurrentes desde los clientes.

A modo de simplificación, se puede asumir que, o bien, todos los clientes y servidor están en el mismo host o todos están en hosts distintos.

Descripción del servidor

El servidor, programado en C/C++, será una aplicación multithreaded que repetirá mensajes entre clientes. Aceptará conexiones iniciadas de los clientes y llevará registro de las mismas. Los mensajes recibidos desde un cliente serán enviados (relayed) a todos los clientes registrados en ese momento. No se llevará registro de los mensajes históricos.

Cuando un cliente desee compartir una imagen suya con otros clientes, el servidor determinará la IP desde la que está conectado el cliente y lo notificará a los otros clientes para que éstos puedan descargar la imagen. Es importante notar que el servidor hace relay de los mensajes, pero no de las imágenes. La transferencia de imágenes es directa entre los clientes.

Luego que un cliente se conecte recibirá mensajes de todos los clientes, así como, sus

mensajes serán copiados a todos los clientes.

Se deberán aceptar conexiones y desconexiones de los clientes en cualquier momento. Las estructuras compartidas por los threads deberán estar correctamente mutuo-excluidas.

El servidor tendrá una interfaz textual mínima, mostrando información de estado elemental. El servidor debe responder a caracteres presionados desde el teclado, mostrando información como:

a – cantidad de clientes conectados

s – cantidad de mensajes enviados

d – cantidad de conexiones totales

f – tiempo (wall time) de ejecución

Cada vez que un cliente envíe un mensaje, éste será copiado por salida standard del proceso servidor (además de a los otros clientes) mostrando la dirección IP origen

Entregable:

Se deberá entregar:

1. Documentación, con énfasis en los protocolos (descripción exhaustiva en idioma Español). Pseudocódigo detallado y comentado de cliente y servidor. Deberá poner énfasis en cómo un cliente obtiene un puerto aleatorio en su máquina y lo hace saber al resto para que lo puedan contactar.
2. Cliente
 1. Código fuente del cliente
 2. Makefile o ant para la compilación de éste (siempre que sea más complejo que `javac Cliente.java`)
3. Servidor
 1. Código fuente del servidor (implementando el pseudocódigo)
 2. Makefile o ant para la compilación de éste (siempre que sea más complejo que `gcc ircserver.cc`)

Insumos:

En la página del curso podrá descargarse el archivo `Cliente.java` con la propuesta inicial de GUI.

Nota

La documentación del obligatorio debe incluirse dentro del archivo de la entrega. La misma debe entregarse como un único archivo tipo PDF de nombre '**informegrupoXX.pdf**'. El mismo deberá respetar la numeración de secciones acorde a los requerimientos anteriores.

Es necesario que en el informe figuren el nombre y la cédula de identidad de cada integrante del grupo. En caso que esto no se cumpla el obligatorio no será corregido, con la consecuente pérdida del curso de sus autores. **NO se aceptarán otros formatos de informe.** (ver <http://www.universidad.edu.uy/odfpdf/>)

have fun!