

Práctica 2:

Chat multiciente

Universidad de Málaga - E.T.S Ingeniería Informática

Desarrollo de Servicios Telemáticos

Antonio J. Galán Herrera

Indicaciones previas	2
Manual de usuario	3
Servidor	3
Cliente	4
Funcionamiento	5
Discusión crítica	6
Bondades	6
Defectos	6

Indicaciones previas

Esta práctica es funcional y cumple los requisitos del enunciado.

Todo el proyecto ha sido ampliamente descrito a través de **comentarios** que dividen el código en secciones dentro de los procedimientos, así como la inclusión de Javadocs para describir la función de dichos métodos. Considero por tanto, que la mejor forma de comprender el funcionamiento es leyendo dichas descripciones de los métodos.

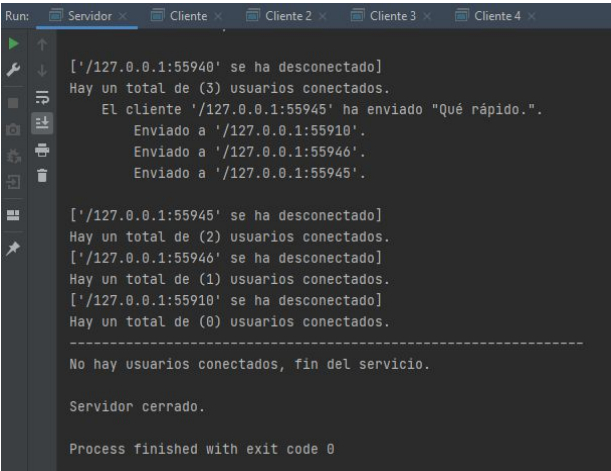
El proyecto se ha creado y construido usando el entorno **IntelliJ**.

Manual de usuario

Servidor

Este proceso puede recibir 1 argumento de entrada: **puerto**; en caso de no recibirlo, se solicitará al usuario que ingrese un puerto por consola.

El servidor es pasivo, por tanto, esa es la única iteración con el usuario. Una vez se establezca un puerto esperará conexiones de clientes e iniciará los procesos de intercambio de mensajes.



```
Run: Servidor x Cliente x Cliente 2 x Cliente 3 x Cliente 4 x
[ '/127.0.0.1:55940' se ha desconectado ]
Hay un total de (3) usuarios conectados.
El cliente '/127.0.0.1:55945' ha enviado "Qué rápido.".
Enviado a '/127.0.0.1:55910'.
Enviado a '/127.0.0.1:55946'.
Enviado a '/127.0.0.1:55945'.

[ '/127.0.0.1:55945' se ha desconectado ]
Hay un total de (2) usuarios conectados.
[ '/127.0.0.1:55946' se ha desconectado ]
Hay un total de (1) usuarios conectados.
[ '/127.0.0.1:55910' se ha desconectado ]
Hay un total de (0) usuarios conectados.

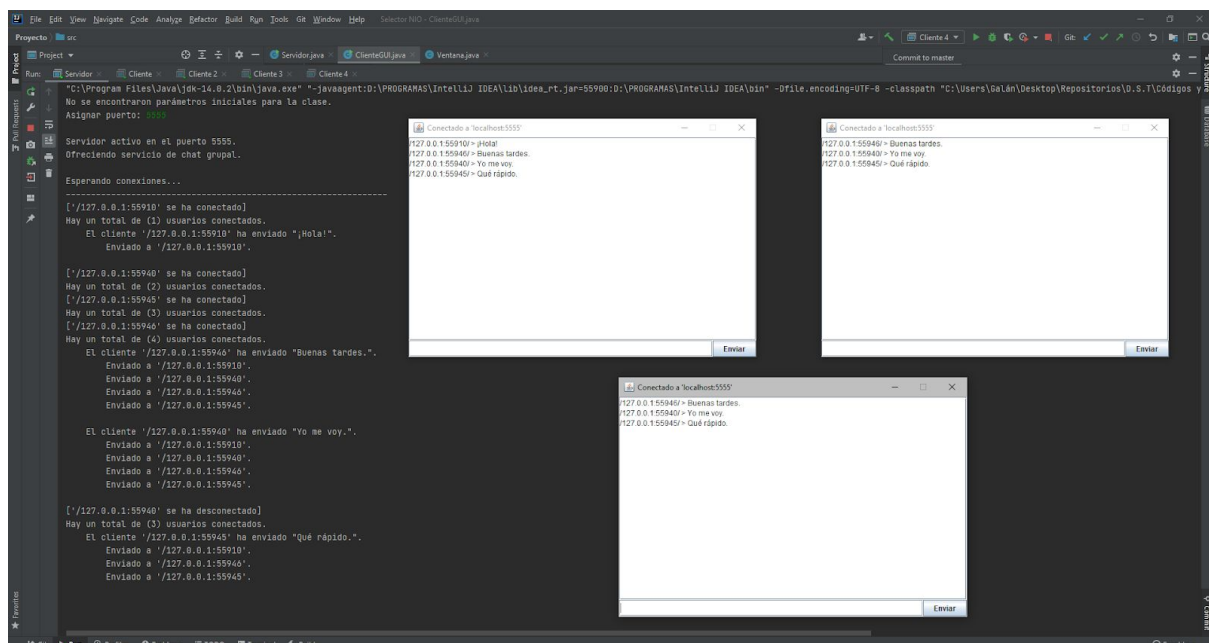
-----
No hay usuarios conectados, fin del servicio.

Servidor cerrado.

Process finished with exit code 0
```

El proceso finalizará una vez todos los clientes se hayan desconectado.

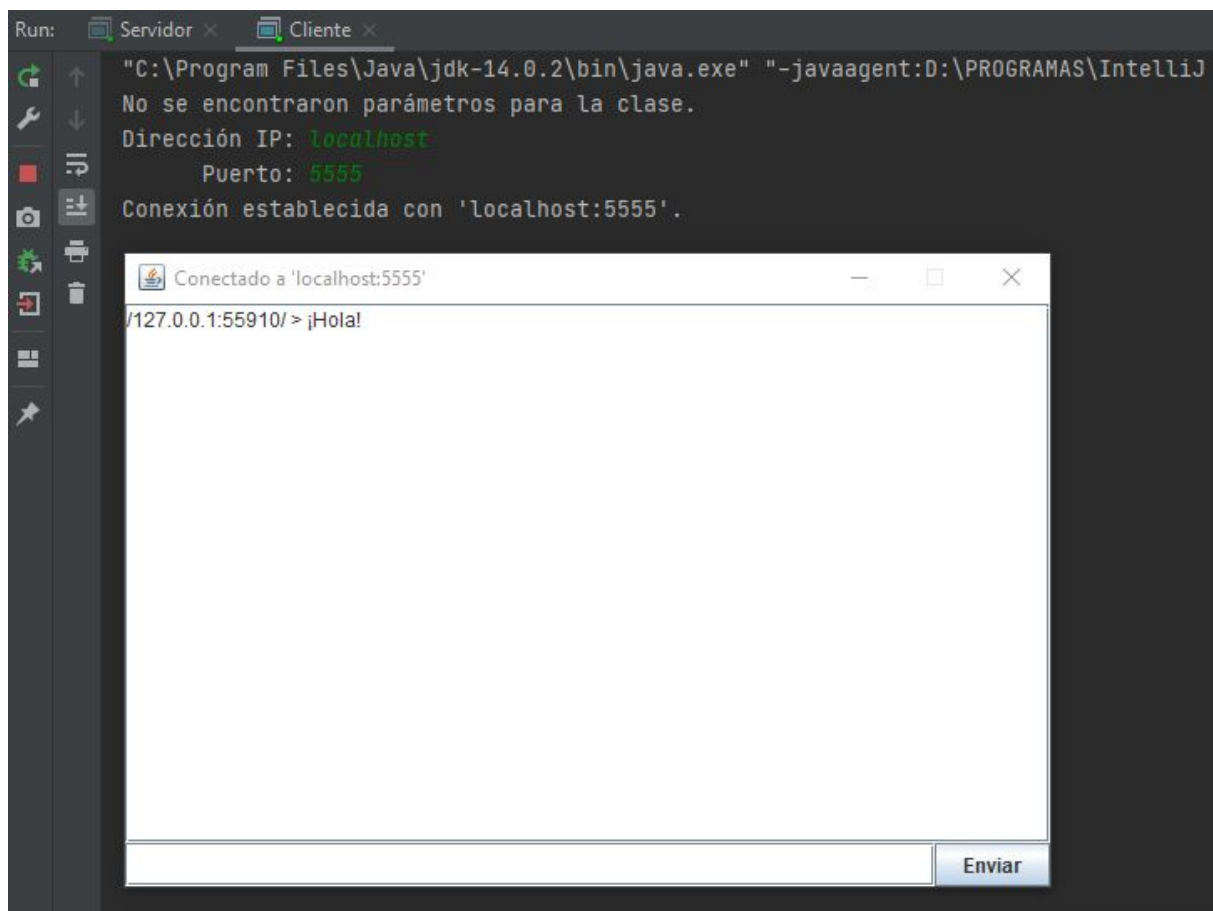
Cuando esto ocurra, el servidor mostrará por consola la información sobre conexiones entrantes y salientes, así como los mensajes recibidos y retransmitidos (como se muestran en la imagen).



Cliente

Este proceso puede recibir 2 argumentos de entrada: **IP** y **puerto**; si se reciben, se establecerá una conexión automáticamente; en caso contrario, se le pedirán al usuario que los introduzca manualmente antes de empezar.

Una vez establecida la conexión, se iniciará una interfaz gráfica GUI en forma de ventana (como se muestra en la imagen). Funciona como cualquier programa de mensajería instantánea, escribiendo el mensaje en la parte inferior y pulsando el botón *Enviar* o bien, pulsando *Enter*; mientras que tanto los mensajes entrantes como una copia del mensaje enviado aparecerán en la parte superior, junto con información del usuario que ha escrito el mensaje.

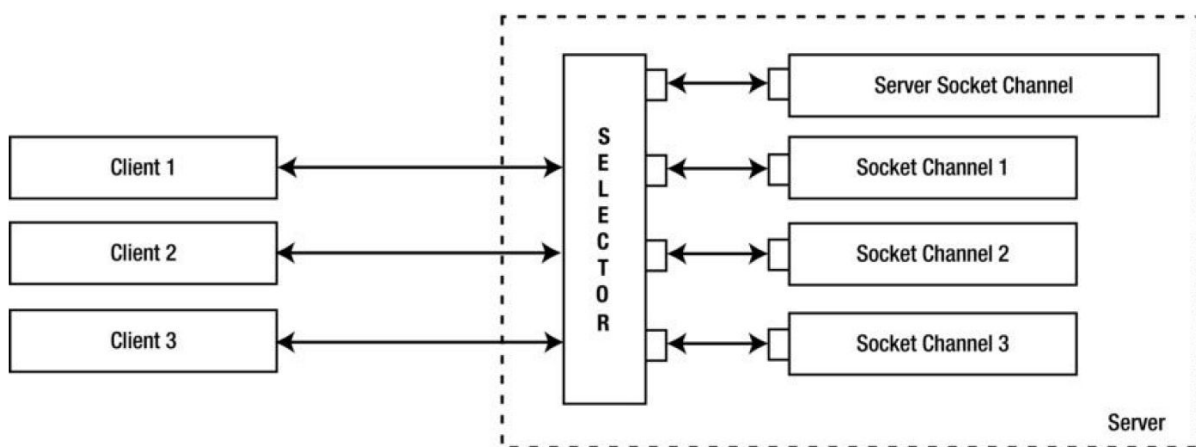


Para finalizar el proceso el usuario solo tiene que cerrar la ventana de la GUI.

Funcionamiento

Tanto el cliente como el servidor necesitarán unos parámetros iniciales asociados a la conexión: el servidor un puerto donde alojar el servicio, y el cliente la dirección del servidor y el mismo puerto.

Una vez añadidos dichos parámetros, el servidor esperará notificaciones, que se distinguirán entre nuevas conexiones o mensajes de clientes; mientras que el cliente creará una GUI inicialmente y desde ese momento se dedicará a enviar y recibir mensajes.



Si un nuevo cliente se conecta, el servidor establecerá un canal de escritura para dicho cliente.

Si se recibe un mensaje, entonces se reenviará por todos los canales disponibles como si fuera broadcast.

Todos los clientes conectados al servidor recibirán el mensaje a través de su canal, incluido el propio usuario que lo envió, ya que de esta forma aparecerá en la GUI.

Discusión crítica

Bondades

Todo el código está altamente comentado (incluido javadocs) y modularizado.

El servidor muestra mucha información en la consola.

Defectos

Podría haberse incluido información adicional en la consola del cliente, al igual que sucede con la consola del servidor y que cuando un cliente se desconectara, se enviara un mensaje informativo en la GUI.