

## PRÁCTICO 2

### CLIENTE Y SERVIDOR CHAT.

#### Propósito

Aprender a programar y extender programas de aplicación que se comunican sobre Internet

#### Generalidades

Desarrolle, ejecute y testee un programa cliente *chat* y un programa servidor *chat*. Luego ejecute ambos programas en computadoras separadas, y verifique que se pueden comunicar.

Luego de haber realizado el laboratorio 1, usted está en condiciones de comenzar con este nuevo laboratorio. El mismo consiste en un cliente *chat* y en un servidor *chat*. Como su nombre lo indica, *chat*, nos va a permitir que ambas partes interactuen como en una conversación normal que se puede establecer entre dos personas.

Una **observacion** que debemos realizar es que en un principio, ambas partes se comportarían de una manera semejante, es decir, establecida la comunicación, tanto el cliente como el servidor se convierten en pares que intercambian datos. Si ambos son iguales, el paradigma cliente servidor se rompería. Esto tiene su explicación y deberá aguardar hasta el laboratorio 3 para la misma.

#### Procedimientos y Detalles

1. Con el código desarrollado en el primer laboratorio, deberá implementar un cliente *chat* y un servidor *chat*. Nuevamente, el servidor escuchará en un puerto determinado y el cliente intentará conectarse con él, en dicho puerto.
2. El servidor *chat* deberá escuchar indefinidamente por comunicaciones pero sólo puede atender una a la vez.
3. Una vez establecida la comunicación entre ambas partes, tanto el cliente como el servidor pondrán enviar mensajes en cualquier momento.
4. Una detalle a tener en cuenta. En esta comunicación, se manejan dos tipos de entradas de datos: tanto de teclado, como de la otra parte. Esto requiere el uso de la herramienta *select* provista por python. Deberá investigar la misma y su funcionamiento.
5. Modifique el cliente y el servidor para que ambos requieran al usuario un nombre (un nombre arbitrario). Incluir el nombre junto con un signo mayor (>) al comienzo de cada línea de texto antes de que sea enviada a través de internet.

#### Obligatorios

6. Modifique el programa para que tenga apariencia de un verdadero chat. Para ello, ambas partes deben de imprimir lo que la otra entidad envía. El lado servidor imprimirá el nombre del cliente, seguido por su mensaje. El lado usuario imprimirá el nombre Servidor, seguido del mensaje obtenido.

Ejemplo:

Si el cliente se llama *Facundo*, y el mensaje recibido por el servidor es *Hola, como estas?*, entonces deberá aparecer por pantalla lo siguiente:

Facundo > Hola, como estas?

Servidor > Bien, y vos?

7. Agregue un temporizador al programa chat para que la conexión TCP sea automáticamente cerrada luego de un minuto de tiempo muerto (es decir, un minuto en el cual ningún carácter es enviado ni recibido). Tenga en cuenta de que este cierre de conexión debe manejarse adecuadamente en ambos extremos para que los sockets creados sean correctamente liberados y queden disponibles para su reuso.