



SAFA

EXAMEN UD3.- PROGRAMACIÓN DE COMUNICACIONES EN RED

Examen PSP

Actividad 1 (2,5 puntos)

Realiza un CHAT UDP (usando multicast).

El **servidor** debe mostrar una pantalla inicial donde tenemos un campo de texto para escribir el mensaje que se enviará a todos los clientes y un textarea donde se van mostrando los mensajes que se van enviando. El botón *Enviar* envía el mensaje escrito a todos los clientes que forman parte del grupo multicast y el botón *Salir* finaliza la ejecución del servidor.

El programa **cliente** pide el nombre al usuario y a continuación muestra un textarea donde se irán visualizando los mensajes que envían el servidor. El botón *Salir* finaliza la ejecución.

Actividad 2 (2,5 puntos)

En el programa servidor, cada vez que se conecte un cliente se creará un nuevo hilo para atenderlo. Se mostrará un TextArea a modo de visor con la dirección IP y el puerto remoto del cliente que se conecta y cuando el cliente se desconecte se debe mostrar un mensaje indicando que se ha desconectado. Además, el servidor generará un número aleatorio entre 0 y 25 al pulsar el botón “Generar”.

En el hilo que atiende al **cliente** recibe un número entero, controlando que si lo que se introduce no es un número, no de un error. En el programa cliente al pulsar el botón *Enviar*, envía el número al servidor. Este comprobará si coincide con el número aleatorio que generó. En caso negativo o positivo, se lo comunicará al cliente de vuelta. El botón *Limpiar* limpia los dos campos y el botón *Salir* envía un * al servidor también finaliza la ejecución.

Actividad 3 (5 puntos)

Crea las siguientes clases con los siguientes atributos, constructores y métodos get y set.

Clase *Barco*:

```
String nombre; //Thousand Sunny
String tripulantes; //Piratas de Sombrero de Paja
String tipo; //bergantín
Integer altura; //56 metros
Integer longitud; //39 metros
```

Clase *Fruta*:

```
String nombre; //Gomu Gomu
String descripcion; /*permite al consumidor obtener
                     así un cuerpo con las propiedades
                     de la goma*/
String apariencia; /*es una violeta fruta esférica,
                     con un tallo que brota de la parte
                     superior, con remolinos que van en
                     un rizado de «S». */
```

Clase *PersonajeOnePiece*:

```
String nombre; //luffy
String rol; //capitan
Fruta frutaDiablo; //Gomu Gomu
Barco nombreBarco; //Thousand Sunny
```

Supongamos que un **cliente** quiere consultar los datos de un personaje de One Piece en **Servidor**.

Utilizando sockets TCP crea un programa servidor que inicialice un array de objetos de tipo *PersonajeOnePiece*. El servidor se ejecutará en un bucle infinito. Cada vez que se conecte un cliente, el servidor le asignará un identificador, este empezará en 1 y se incrementará en 1 según se van aceptando conexiones de clientes. Nada más conectarse el cliente, el servidor le enviará el identificador que le ha correspondido. Este servidor puede atender a múltiples clientes utilizando hilos.

Crea un programa **cliente** que una vez conectado al cliente muestre el identificador que le ha correspondido. El cliente introducirá por teclado el *nombre del personaje* que desea consultar, se realizará la lectura en un proceso repetitivo hasta que se introduzca por teclado un *. Se enviará al servidor el *nombre* a consultar. El servidor le devolverá un objeto *PersonajeOnePiece* con los datos solicitados. Si el *personaje* no existe, también le devolverá un objeto *PersonajeOnePiece*, con datos que indiquen que el personaje no existe. El cliente debe visualizar todos los datos del personaje y, si lo desea, los datos de su fruta y barco.

Controlar todos los posibles errores y cuando un cliente se desconecte mostrar también un mensaje indicándolo.