**Indicaciones de entrega**

Una vez realizada la tarea elaborarás un único fichero **.zip** donde figuren las respuestas correspondientes. El envío se realizará a través de la plataforma de la forma establecida para ello, y el archivo se nombrará siguiendo las siguientes pautas:

**apellido1\_apellido2\_nombre\_SIGxx\_Tarea**

Asegúrate que el nombre no contenga la letra ñ, tildes ni caracteres especiales extraños. Así por ejemplo la alumna **Begoña Sánchez Mañas para la primera unidad del MP de DAW**, debería nombrar esta tarea como...

**sanchez\_manas\_begona\_PSP\_Tarea01**

**Cada ejercicio ha de desarrollarse en un documento distinto, siguiendo los nombres de ej1, ej2,..**

**Ejercicios**

1. **Crear dos clases de Java (cliente y servidor)** de forma que el servidor abra un puerto en escucha y quede a la escucha por él para recibir datos del cliente.

* El cliente pedirá una cadena por teclado y la enviará al servidor.
* El servidor la recibe, la muestra por consola y hará un echo de nuevo al cliente (“SERVER ECHO: \*cadena del cliente\*”)
* El cliente la recibe y muestra por consola.

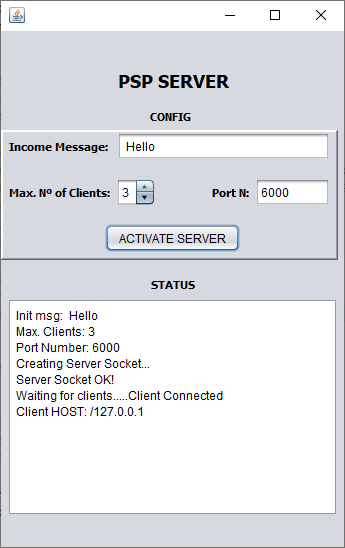
1. **Crea un Servidor con JFrame como el de la imagen.**

* Crea la interfaz gráfica y almacena los parámetros en variables (cuidado los tipos)
* Utiliza el cliente del ejemplo anterior para conectarte comentando la parte de flujo de datos
* Primero haz que se conecte un sólo cliente (sin envío de datos) ¿Qué ocurre? ¿Porqué?
* Realiza un bucle para aceptar el Nº de conexiones configuradas y verifica que una vez conectados tantos clientes muestra toda la información.

**Ayuda:**

int maxClient=Integer.parseInt(numMax.getValue().toString());

textStatus.append("Init msg: "+message+"\n");



1. **Ejercicio (Sin JFrame):** Crea un Servidor que genere un hilo para cada cliente que se conecte, muestre por pantalla el número y el nombre (se lo enviará el cliente y le envíe su Nº de cliente.

El programa cliente será similar al que hemos estado utilizando, y deberá mostrar por consola la siguiente información:

***Conectando con servidor IP: xxxxx, Puerto: xxxx …***

***Conectado!***

***Enviando nombre…***

***OK!***

***Recibiendo N de cliente…***

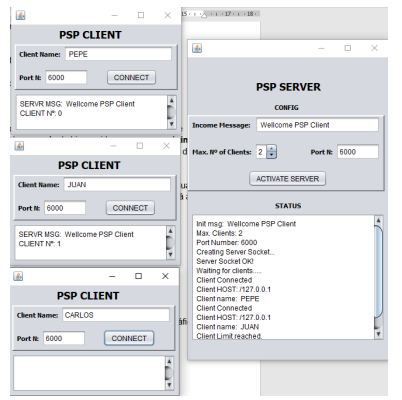
***Cliente conectado Nº xx***

***cerrando conexión…***

***fin conexión.***

1. **Realiza con una interfaz visual en Java dos aplicaciones: una cliente y otra servidor de forma que:**
   1. El **servidor** permita configurar Un mensaje de bienvenida y el puerto de conexión.
   2. Además debe mostrar información del proceso y de los clientes conectados. A cada cliente le asignará una ID. Cada cliente debe ser atendido mediante un Thread de forma concurrente

* EL programa **cliente**, permitirá ingresar un nombre de cliente y el número de puerto. Cuando se conecte **recibirá** del servidor el **mensaje** de bienvenida y el **ID** de cliente y le enviará al mismo su nombre de cliente.
* Añade **solicitud de servicios** desde el lado cliente (por ejemplo obtener direcciones web, fecha/hora, alguna coordenada, etc.) El cliente detectará la solicitud y le mandará lo solicitado. (**Negociación** C-S por comandos o similar)
* Añade en el cliente la **dirección IP** (o nombre de dominio) y corre las aplicaciones en **máquinas separadas.**



1. Usa un constructor para crear un **datagrama UDP** de salida a la IP 192.168.1.5 puerto 12345 con el mensaje “Paquete de prueba 1” (Utiliza el método getBytes() de String para obtener el buffer.

