# Reporte de la práctica 1

## Desarrollo

### Archivos Principales

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

La práctica cuenta con esas clases, las más importantes son, Cliente, Interfaz y Servidor.

De manera general, se puede decir, que la práctica se realizo con Sockets de Datagrama y con un objeto de tipo lista que guardaba los datos de cada cliente que se fuera registrando, dichos datos son: nombre, mensaje y puerto del Socket de Datagrama.

### Clase Cliente.java

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

La clase cliente, primero envía el nombre del usuario al Servidor, como se ve en la línea 27.

Cuenta con 3 hilos, un HILO que siempre está escuchando un mensaje privado (línea 28), otro que está escuchando un mensaje multicast (línea 29) y uno último que siempre está listo para enviar mensajes (línea 33).

### Clase Servidor.java

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

La clase servidor, tiene 2 hilos. Un hilo que siempre está escuchando los datos de cada cliente y otro que está preparado para hacer una búsqueda del destinatario del mensaje

### Clase SocketEscuchaMulticast.java

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto, Aplicación

Descripción generada automáticamenteEsta clase lo que hace es conectarse a un grupo de direcciones multicast, para que aquel cliente que mande un datagrama a esa dirección, se lo envíe a todos los sockets de datagrama que están escuchando y que por obvio pertenecen a esa dirección de grupo multicast.

### Clase SocketEscucha.java

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Esta clase prácticamente funciona igual que SocketEscuchaMulticast.java, sólo que aquí no se envían los datos a una dirección multicast, sino a una dirección única que se buscará en un futuro.

Además de que un método extra, que se llama enviaDatosAlServidor

### Clase DatosUsuario.java

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Esta clase, sólo se ocupa como auxiliar en la creación de la lista de Usuarios.

### Clase Búsqueda.java

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Está clase sirve para buscar los datos de un usuario al cual se le enviará el mensaje, luego, esa información va a otro Socket de Datagrama, este último Socket envía la información al Cliente que siempre está escuchando mensajes

### Clase IPyPuertoAleatorio.java

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Este Socket del lado del servidor, recibe los datos de un usuario y los guarda en una lista que posteriormente será ocupada para buscar el usuario destinatario del mensaje.

### Clase ListaUsuarios.java

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente  
Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

La clase lista usuarios tiene una lista del tipo DatoUsuarios, es útil para buscar el puerto del Socket de Datagrama del Cliente Destino.