

## Sistema de mensajería instantánea (*Méseller*)

### Concepto

*Méseller* es una aplicación de mensajería instantánea entre usuarios que contará con acceso anónimo (en caso de solicitarlo) así como de creación de salas de chat creadas por los propios usuarios. Por supuesto, se incluirán las funcionalidades clásicas de este tipo de sistemas (comunicación unilateral entre usuarios, agenda de contactos, etc).

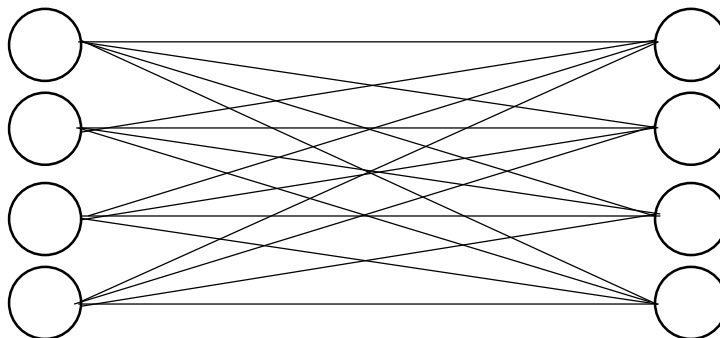
Se hace hincapié en la importancia de la seguridad de los usuarios y del sistema como tal, por lo que se escogerá un lenguaje que responda a estas características.

La idea de la realización de un sistema de mensajería instantánea usando erlang es aprovechar su potencial de creación y gestión de procesos (procesos erlang) así como su facilidad a la hora de la comunicación entre éstos. Por debajo de estos puntos, aunque no menos importante, encontramos su usabilidad frente a otros lenguajes como C o Java para aprovechar la recursión y el tipo de variables que ofrece.

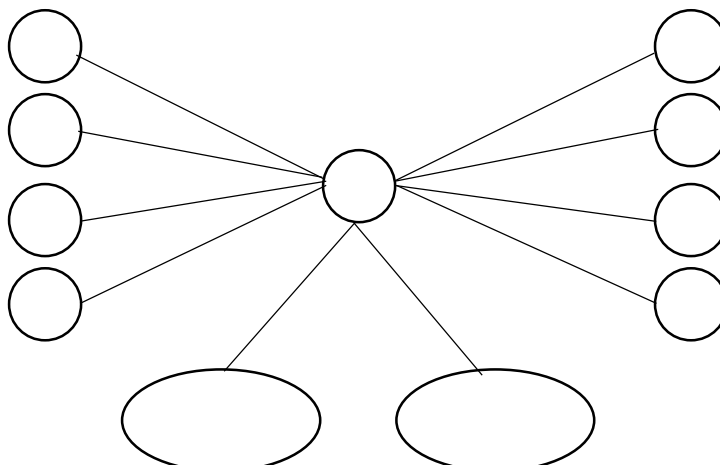
### Esquema de conexiones

En un principio, se podrían realizar las comunicaciones directamente entre usuarios, existiendo sólo los procesos clientes, pero esto supone dos grandes problemas:

- Falta de control y gestión de los usuarios y las comunicaciones
- Comunicaciones administradas de forma independiente, pudiendo llegar a colapsar el proceso cliente



Por estos motivos, la existencia de un proceso intermedio, servidor, facilitará tanto la gestión como las comunicaciones. Cualquier usuario se comunicará con cualquier otro a través del servidor, nunca de forma directa, de este modo, se añade una capa de seguridad y control. Además, la existencia de este servidor posibilita la creación de salas de chat para varios usuarios.



En vista de lo anterior, se podría pensar que será el servidor el que pueda colapsar pero será trabajo de los administradores asegurarse de que este proceso está activo y da servicio a todos los usuarios.

### **Interfaz**

La interfaz de la aplicación se realizará a través de Java, interactuando con la aplicación cliente a través de la API para Java de erlang JInterface.