

TRABAJO PRACTICO SO1

Franco García Cillero G-5238/8

Dante Noguera N-1160/6

Overview

Cuando un cliente se conecta al servidor **dispatcher** lanza **psocket** que se encarga de parsear los comandos que ingrese el cliente, preguntar a **pbalance** por el nodo con menos carga y lanzar **pcomando** en dicho nodo (el criterio para determinar el nodo con menos carga es la cantidad de puertos de erlang abiertos que tiene el nodo esto indica la cantidad de clientes conectados al nodo/servidor). Además lanza **sendUpdate** que envía actualizaciones del servidor al usuario. **pcomando** ejecuta los comandos en el servidor, **usersManager** se encarga de los comandos relacionados con registro de usuarios, **gamesManager** se encarga de los comandos relacionados con los juegos.

Decisiones tomadas

- Un usuario se puede registrar una única vez. Si intenta registrarse de nuevo será bloqueado.
- Un usuario no puede jugar consigo mismo.
- Cuando un usuario crea una partida no puede jugar hasta que haya un contrincante.
- Cuando una partida termina se resetea.
- Una partida se puede abandonar en cualquier momento, la misma se resetea.
- Un usuario no puede observar más de una vez una partida.
- Un jugador no puede observar la partida en la que juega.
- Cuando un jugador abandona la partida, y queda otro jugador, esta se resetea. Si ambos abandonan la partida es destruída.

Para probar el servidor

1. Crear un nodo: `erl -name name@ip`.
2. Compilar el archivo `server.erl`: `c(server)`.
3. Correr la función `init`: `tp:init(Port, Node)` donde `Node` es el nombre de un nodo al que se quiera conectar.
4. Conectar a los servidores con `telnet` para hacer de cliente.

Ejemplo:

1. `erl -name nA@127.0.0.1`

2. `erl -name nB@127.0.0.1`
3. En nA: `c(server), server:init(8000, 'nB@127.0.0.1')`.
4. En nB: `c(server), server:init(8080, 'nA@127.0.0.1')`.
5. `telnet localhost 8000` (cliente de nA).
6. `telnet localhost 8080` (cliente de nB).