

**Disseny**

El Servidor consta de dues classes, InetNode i Servidor. InetNode és una classe genèrica que facilita l'ús dels missatges. Els camps que es guarden són:

- nom: és el nom que li donem a un node
- socket: es el socket, pel qual enviarem les dades
- adreça: es la direcció ip del node
- port: port és per saber el port que té
- estatus: és una cadena de caràcters que conté l'estat en el que està el node
- expansió: camp d'expansió de protocol

El camp estatus en el cas dels servidor pot ser: desconnectat i servidor.

En el cas dels client pot ser: desconnectat, connectat, identificat i registrat.

Segons l'estat en el que es troben, accepta com a vàlides una sèrie de comandes diferents.

HELO provoca passar de connectat a identificat

REGISTER (amb un nom únic) passa de identificat a registrat.

Aquestes funcions estan implementades en el Servidor ,en lloc de en InetNode.

El camp expansió en cara que no es fa anar és per afegir funcionalitats que ens puguin interessar. La idea és que el servidor al enviar el OK (o text equivalent) de resposta utilitzi un nom determinat el qual doni a entendre al client que disposa d'una sèrie de funcions extra (ex: tots els clients només llegeixen el 200 per verificar-ne la correctesa, es en aquest missatge posterior on hi hauria el text identificador). D'aquesta manera es podria implementar funcions que passessin la llista de clients connectats al client que s'acaba de connectar o per part del servidor d'avisar si un client a desconnectat.

El servidor disposa dels següents camps:

- clients: clients és una llista de InetNode
- màxim de clients: Quantitat estipulada de InetNode que podem tenir / escoltar

El servidor està posat dins d'aquesta llista de clients, es troba en la primera posició. Tal com esta codificat en aquests moments només podem tenir un InetNode del tipus servidor (sempre es troba en la primera posició, ja que és prioritari). Està fet d'aquesta manera per si ens interressa fer que el client també pogués parlar amb els clients.

**Excepcions**

Es dona el cas que si el port s'ha tancat malament o hi ha una altra aplicació que n'ha fet un binding l'aplicació falla, ja que no pot obrir el port especificat

**Casos especials**

No es pot tenir més de 10 clients connectats. No és el limit real, però com que no ens posarem a connectar una quantitat molt elevada de clients, a més de considerar que hi haurien clients que disposarien d'una prioritat inferior, per la quantitat de sockets que podríem escoltar e posat aquest límit. No falla tant sols que el servidor ens tancarà el socket.

Un altre cas que he considerat es que si el client envia un missatge que no contenen res es que ha quedat buclat i per tant en tanco la connexió.