

# Speaker: Proposta

Jordi Bartrolí

Héctor Mañosas

Ramon Xuriguera

Març 2010

## 1 Resum de la proposta

El projecte que pretenem desenvolupar per l'assignatura de PXC és una aplicació per escoltar cançons via *streaming* en una xarxa *peer-to-peer*; que permeti escoltar temes musicals buscant per artista, nom de la cançó o àlbum.

Actualment existeixen nombrosos serveis d'*streaming* de música similars: *Last.fm*, *Spotify*, *Grooveshark*, etc. De tots aquests, només *Spotify* fa ús de les tècniques P2P per repartir el tràfic entre els usuaris de la xarxa i tot i així, també s'ajuda de servidors d'*streaming*. També es poden trobar altres aplicacions com ara *Despotify*, però només són eines generades amb enginyeria inversa que per sota utilitzen els servidors i el protocol d'*Spotify*.

Creiem que fa falta una alternativa. La nostra proposta té els següents aspectes diferenciadors:

- 100% P2P
- Open Source (Llicència GPL v.3)
- Lliure de publicitat
- Possibilitat de descarregar música

## 2 Objectius tecnològics

Una de les tasques que hem realitzat abans de redactar aquesta proposta i a la que hem dedicat força temps ha estat la recerca de tecnologies i treball existent que ens poden ajudar a aconseguir els nostres objectius. Per implementar la nostra aplicació en el període de temps requerit serà necessari utilitzar les eines següents:

- Biblioteques d'*streaming*
- Reproductor d'àudio
- Framework P2P com ara JXTA
- Biblioteques per obtenir la informació ID3 dels arxius d'àudio

Altres tecnologies que utilitzarem:

- RMI over SSL
- Servlets

- SQL
- XSLT
- DHT

### 3 Descripció detallada de la proposta

La nostra aplicació està pensada per funcionar en un entorn com el de la figura 1. Per una banda tenim un servidor amb una base de dades accessible tant a través del navegador (amb *servlets*) com RMI. Aquest servidor ens permetrà tenir un control dels usuaris de l'aplicació i generar estadístiques.

Per l'altra, tindrem una xarxa de *peers* de diferents tipus:

- *Rendezvous peers*: Nodes de la xarxa fixes que permetran que els nous nodes en puguin conèixer d'altres. En principi, aquesta funció la farà el mateix servidor que conté la base de dades.
- *Edge peers*: Nodes dels usuaris. Seran màquines que compliran funcions tant de client com de servidor per tal d'intercanviar cançons.
- *Peers mòbils*: Nodes dels usuaris que s'executen en un dispositiu mòbil. Només faran de client, és a dir, només podran rebre cançons. Hem pres aquesta decisió perquè es tracta de dispositius molt més limitats en qüestions de processador i capacitat d'emmagatzemament.

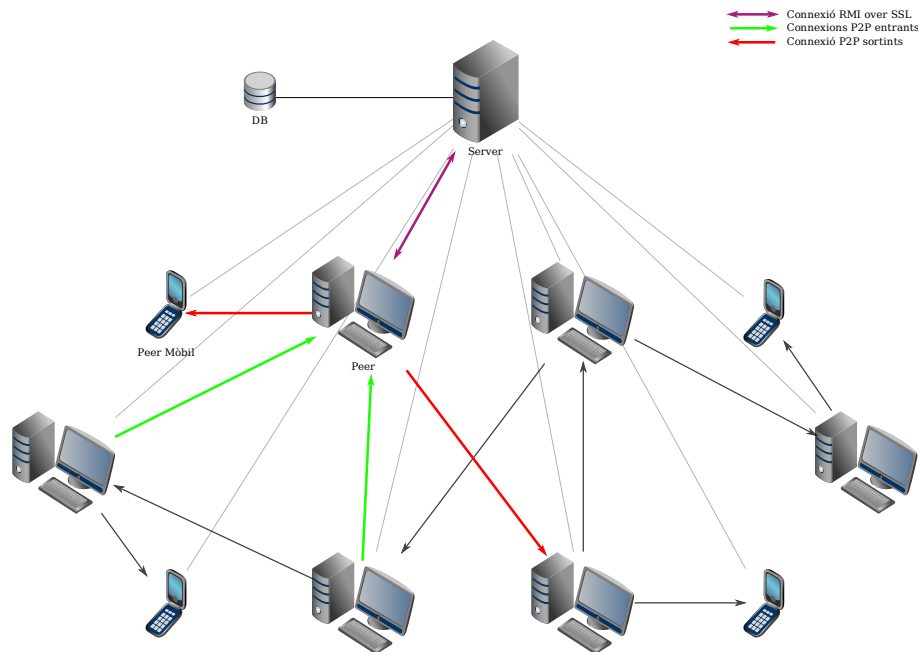


Figura 1: Topologia de la xarxa

### 3.1 Flux de la informació

Un dels problemes que creiem que hi pot haver a la pràctica és que els *peers* descarreguin molta més música de la que pugen. De fet, un dels factors que influeix en això és el fet que els *peers* mòbils només facin de client. Creiem que obligar als usuaris a registrar-se pot ser una solució. D'aquesta manera, podem controlar qui pot utilitzar l'aplicació mòbil i qui no. A més, també ens servirà per mostrar estadístiques d'ús de l'aplicació.

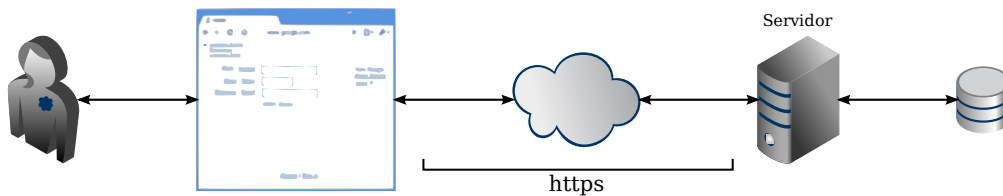


Figura 2: Registre d'usuaris

Per tant, el primer pas per a que un usuari pugui utilitzar la nostra aplicació serà registrar-se des de la nostra pàgina web. Com que les dades que s'enviaran són privades, caldrà que la connexió sigui segura. Un cop registrat, l'usuari es pot descarregar l'aplicació i instal·lar-la.

Abans de poder començar a escoltar música, l'usuari s'haurà d'autenticar. Com en el cas anterior, el nom d'usuari i clau s'enviaran al servidor de forma segura, tot i que aquí s'utilitzarà RMI. Si la informació és correcta, el servidor li donarà permís per començar a utilitzar l'aplicació.

Com que el servidor també serà un *rendezvous peer*, li facilitarà una llista de peers per tal que es pugui unir a la xarxa. A partir d'aquí, l'usuari ja estarà connectat i podrà començar l'intercanvi de fitxers d'àudio.

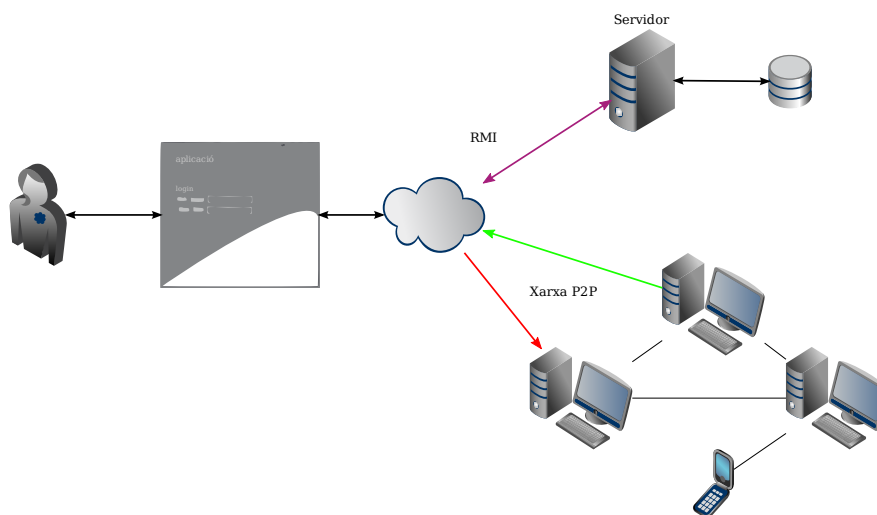


Figura 3: Autenticació des de l'aplicació

Per poder escoltar música farà falta que els peers comparteixin els fitxers que tenen a la seva màquina local i puguin buscar-ne d'altres. Per facilitar la cerca, l'aplicació tindrà una biblioteca de música que consistirà en un fitxer XML amb un llistat dels fitxers disponibles i la seva informació, per exemple: el nom de la cançó, artista, hash del fitxer, etc.

A més, ha de ser possible trobar fitxers de màquines que estan a la xarxa, però que no coneixem directament. Per fer-ho utilitzarem la tècnica de *Distributed Hash Tables*. La figura 4 mostra aquesta idea. Les fletxes blaves indiquen la cerca d'un fitxer als *peers*. Encara que el fitxer estigui a una màquina que en principi no coneixem directament, la descàrrega sí que es fa de forma directa (fletxa verda).

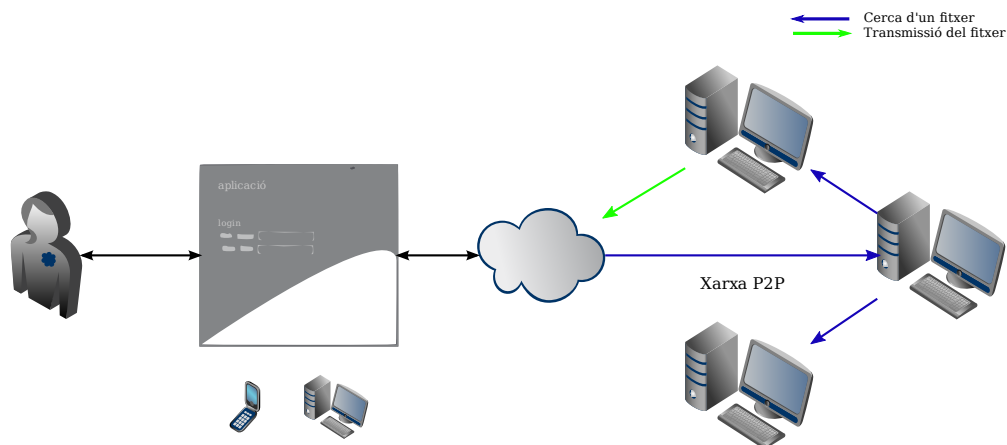


Figura 4: Cerca de cançons

El servidor emmagatzemarà estadístiques de la música que estan escoltant els usuaris. Aquesta informació també l'enviarà l'aplicació utilitzant RMI. Per consultar-la caldrà fer-ho des de la web, on hi haurà disponibles estadístiques tant per usuari com globals. La informació s'extreurà de la base de dades en XML i es convertirà a HTML amb un full d'estil XSL, tal i com es pot veure a la figura 5.

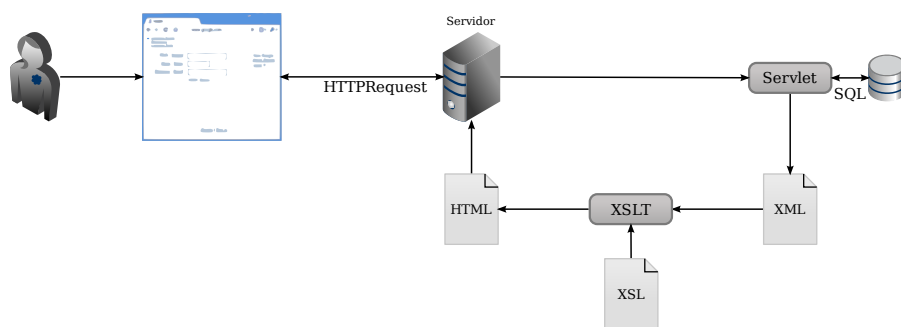


Figura 5: Consulta d'estadístiques des de la web

### 3.2 Relació de la proposta amb l'assignatura

Creiem que la nostra proposta encaixa molt bé en l'àmbit de l'assignatura. Per una banda utilitzarem algunes de les tecnologies que hem vist durant les pràctiques: RMI i Servlets. Per l'altra, hi ha una bona part de treball que, tot i no haver-lo vist a classe, està molt relacionat amb el tema de xarxes: l'streaming i el P2P.

## 4 Arquitectura

Els següents diagrames mostren a grans trets, quina serà l'arquitectura de capes dels *peers* de la xarxa i del servidor. Els *peers* mòbils seran com els que es mostren al primer diagrama, però sense la possibilitat de fer de servidor P2P.

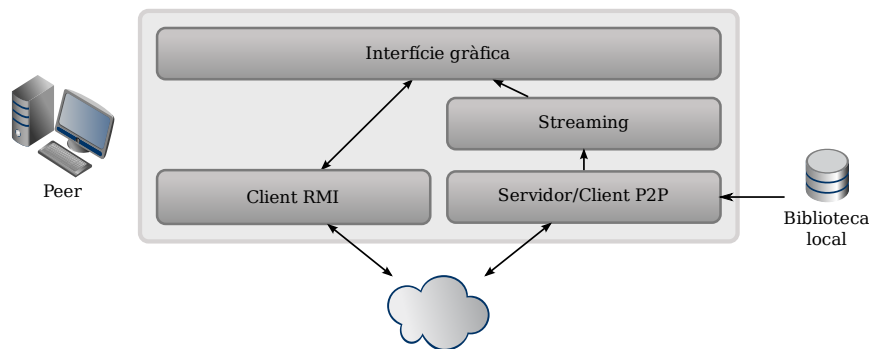


Figura 6: Arquitectura dels *peers* d'escriptori

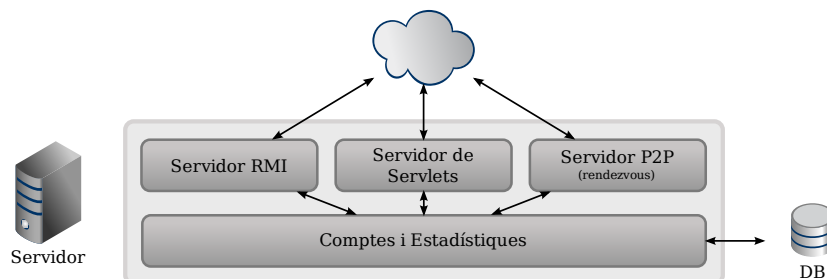


Figura 7: Arquitectura del servidor

## 5 Paquets de treball

A continuació llistem els paquets de treball principals que caldrà realitzar per completar la nostra aplicació. Per cadascun d'ells en detalllem els objectius, una descripció i quin ha de ser el resultat. El diagrama de Gantt que acompanya aquest document mostra la duració estimada i a qui s'ha assignat cada paquet.

### 1. Recerca d'informació

- Objectius: Decidir quines tecnologies cal utilitzar i comprovar que ens poden ajudar a aconseguir els propòsits de l'aplicació.
- Treball a realitzar: Obtenir informació i aprendre a utilitzar les tecnologies.
- Resultat: Llistat de tecnologies i proves de concepte.
- Assignat a: Tots

### 2. Disseny de l'esquema de la base de dades

- Objectius: Crear un diagrama amb l'arquitectura de la base de dades.
- Treball a realitzar: Acordar amb tots els membres de l'equip les característiques necessàries de la base de dades i realitzar un diagrama d'arquitectura.
- Resultat: Un diagrama clar i complet.
- Assignat a: Tots

### 3. Disseny de la interfície Web

- Objectius: Decidir com estructurar la informació i com presentar-la.
- Treball a realitzar: Consultar altres pàgines per veure com solucionen el problema de presentar les dades de forma clara. Fer prototips de paper. Codificar el disseny en XHTML i CSS.
- Resultat: Fitxers HTML i CSS amb la interfície de la web.
- Assignat a: Héctor i Ramon

### 4. Registre d'usuaris i consulta d'estadístiques

- Objectius: Poder crear comptes d'usuari des de la pàgina web i consultar les estadístiques.
- Treball a realitzar: Decidir la informació necessària a demanar als usuaris. Estudiar com enviar les dades de forma segura. Crear dos *servlets*.
- Resultat: *Servlets* per a la creació de nous comptes d'usuari i per a l'obtenció d'estadístiques.
- Assignat a: Héctor

### 5. Creació de la xarxa P2P

- Objectius: Crear un parell de programes client i servidor que es puguin connectar entre ells.
- Treball a realitzar: Crear l'aplicació client, l'aplicació servidor i tenir en compte la necessitat de *rendezvous peers* per comunicar peers disponibles quan algú entra a la xarxa.
- Resultat: Programes client i servidor que permetin crear una xarxa P2P on un peer nou pot arribar a connectar-se amb altres.

- Assignat a: Ramon

## 6. Multithreading client i servidor P2P

- Objectius: Executar en la mateixa màquina tant el servidor com el client P2P.
- Treball a realitzar: Investigar sobre com funcionen els *threads* a Java i implementar un programa que executi el codi del client i el del servidor en fils d'execució diferents.
- Resultat: Una única aplicació que faci de client i servidor P2P utilitzant threads diferents.
- Assignat a: Ramon

## 7. Creació de la biblioteca local

- Objectius: Crear una biblioteca dels fitxers dels quals es disposa a la màquina local i que podran ser compartits amb la resta de peers.
- Treball a realitzar: Decidir quina informació cal fer disponible als usuaris de la xarxa. Trobar els fitxers d'àudio d'un directori. Extreure'n la informació (ID3). Resoldre possibles inconsistències (e.g. cançons duplicades, etc.)
- Resultat: Un fitxer XML amb la descripció de la biblioteca.
- Assignat a: Jordi

## 8. Interfície Gràfica

- Objectius: Crear interfície d'usuari intuïtiva per a la cerca i reproducció de cançons.
- Treball a realitzar: Buscar informació sobre com crear interfícies en Java, fer prototips i implementar la interfície.
- Resultat: Conjunt de classes Java que conformen la interfície de l'aplicació.
- Assignat a: Jordi i Ramon

## 9. Adaptació a Android

- Objectius: Adaptar l'aplicació d'escriptori per utilitzar-la en mòbils Android.
- Treball a realitzar: Buscar informació com desenvolupar aplicacions Java per Android, dissenyar prototips de la interfície gràfica a Android, eliminar el codi de característiques que no s'ofereixen pels *peers* mòbils.
- Resultat: Conjunt de classes Java que conformen l'aplicació d'Android.
- Assignat a: Jordi

## 10. Cerca de cançons dins de la xarxa P2P

- Objectius: Oferir un servei als peers de la xarxa per tal de localitzar cançons d'altres peers.
- Treball a realitzar: Buscar informació sobre com utilitzar DHT amb el framework JXTA. Implementar.

- Resultat: Conjunt de classes Java que ofereixen aquesta funcionalitat.
- Assignat a: Ramon

#### 11. Transmissió de fitxers

- Objectius: Transmetre els fitxers d'audio d'un *peer* a un altre.
- Treball a realitzar: Buscar informació sobre com utilitzar els *sockets* del framework JXTA. Implementar.
- Resultat: Conjunt de classes Java que ofereixen aquesta funcionalitat.
- Assignat a: Ramon

#### 12. Autenticació RMI i enviament d'estadístiques

- Objectius: Permetre l'autenticació d'usuaris des de les aplicacions d'escriptori i mòbil. I permetre l'enviament d'estadístiques d'ús.
- Treball a realitzar: Estudiar com fer invocacions remotes segures (e.g. RMI over SSL) i implementar un client i un servidor que permetin autenticar usuaris i enviar estadístiques de l'aplicació (e.g. cançó que s'està escoltant en un moment donat).
- Resultat: Conjunt de classes Java que permeten autenticar i enviar estadístiques.
- Assignat a: Jordi i Héctor

#### 13. Tests finals

- Objectius: Testejar l'aplicació per assegurar-nos que funcioni.
- Treball a realitzar: Fer proves d'instal·lació, ús i desinstal·lació de les aplicacions d'escriptori i mòbil.
- Resultat: Aplicacions testejades.
- Assignat a: Tots

#### 14. Documentació

- Objectius: Generar la memòria del projecte i manual d'ús per tal que un nou usuari sàpiguí utilitzar l'aplicació.
- Treball a realitzar: Escriure documentació. Dibuixar diagrames.
- Resultat: Manual d'usuari i la memòria del projecte.
- Assignat a: Tots

## 6 Pla de riscos

A continuació detallem els riscos que ens poden sorgir durant la implementació del projecte. Per cadascun d'ells especifiquem la probabilitat, l'impacte sobre el projecte i què farem si realment ens afecta. Com que estem molt condicionats per la data d'entrega del projecte, hem intentat que el pla B de cada risc no modifiqui el diagrama de Gantt per tal de no afectar les tasques successores.



1. No trobem cap manera de cercar arxius amb DHT

- Probabilitat: 0.6
- Impacte: 0.4
- Pla B: Si arribat el dia 19 d'abril no aconseguim tenir-ho funcionant, farem que les cerques es facin consultant directament els fitxers XML de biblioteca.
- **Hazard:**  $0.6 \cdot 0.4 = 0.24$

2. No aconseguim fer funcionar els threads de Java

- Probabilitat: 0.3
- Impacte: 0.5
- Pla B: Si arribat el dia 7 d'abril no aconseguim tenir-ho funcionant, farem que les cerques es facin consultant directament els fitxers XML de biblioteca.
- **Hazard:** 0.15

3. Problemes al transferir cançons amb JXTA

- Probabilitat: 0.3
- Impacte: 0.5
- Pla B: Si arribat el dia 29 d'abril no funciona, utilitzarem RMI..
- **Hazard:** 0.15

4. Problemes amb les llicències d'utilització de codecs per reproduir MP3

- Probabilitat: 0.2
- Impacte: 0.6
- Pla B: Si arribat el dia 5 d'abril no hem aconseguit integrar la reproducció dins de l'aplicació, utilitzarem un reproductor extern.
- **Hazard:** 0.12

5. No tenim temps de crear les pàgines d'estadístiques

- Probabilitat: 0.5
- Impacte: 0.2
- Pla B: Si el 22 d'abril no hem pogut implementar aquesta funcionalitat, la deixarem per la pròxima versió.
- **Hazard:** 0.10

6. Transformació de les dades de la DB a XML

- Probabilitat: 0.2
- Impacte: 0.3

- Pla B: Si el 15 d'abril encara no ho hem pogut implementar, transformarem les dades directament a HTML.
- **Hazard: 0.06**

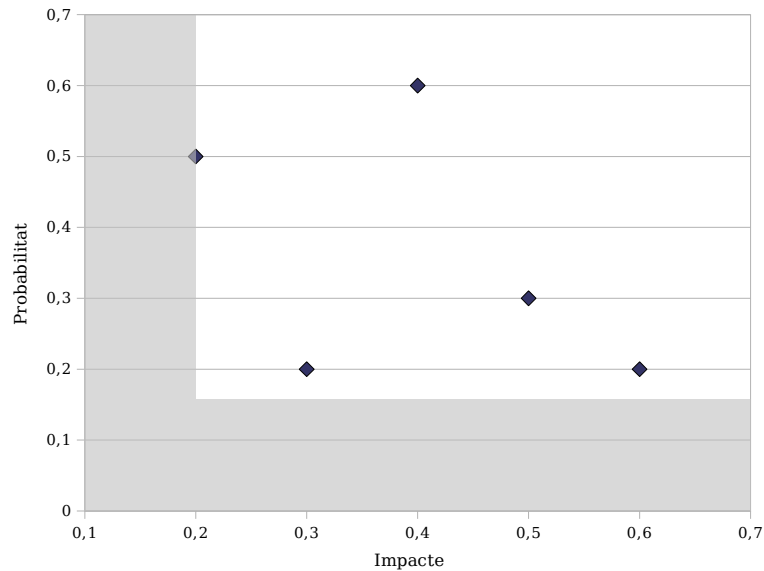


Figura 8: Gràfic de riscos

## 7 El grup de projecte

Tal i com es pot intuir a partir del diagrama de Gantt, els rols dels membres de l'equip són:

- Jordi Bartrolí: Streaming i dispositius mòbils
- Héctor Mañosas: Base de Dades i comunicació amb el servidor
- Ramon Xuriguera: Xarxa P2P

Per desenvolupar el projecte utilitzarem un repositori SVN que es pot trobar a <http://code.google.com/p/pxc2>. Per organitzar-nos, utilitzem l'eina *Planner* amb la qual podem indicar l'estat de cada tasca i reassignar-la a algun altre membre del grup.