



UNIVERSIDAD DE JAÉN
Escuela Politécnica Superior (Jaén)

Proyecto Fin de Carrera

OL-Radio UJA. Ampliación de funcionalidades

Alumno: David Parras Armenteros

Tutores: Dra. Dña. Macarena Espinilla Estévez

Área: Arquitectura de Tecnología y Computadores

Dpto: Informática

Febrero, 2011



UNIVERSIDAD DE JAÉN
Escuela Politécnica Superior (Jaén)

Proyecto Fin de Carrera

OL-Radio UJA. Ampliación de funcionalidades

Alumno: David Parras Armenteros

Tutores: Dra. Dña. Macarena Espinilla Estévez

Área: Arquitectura de Tecnología y Computadores

Dpto: Informática

Febrero, 2011



Universidad de Jaén
Escuela Politécnica Superior de Jaén
Departamento de Informática

Dra. D^a. Macarena Espinilla Estévez, tutora del Proyecto Fin de Carrera titulado: OL-Radio UJA. Ampliación de funcionalidades, que presenta D. David Parras Armenteros, autoriza su presentación para defensa y evaluación en la Escuela Politécnica Superior de Jaén.

Jaén, Febrero 2011

El Alumno:

La Tutora:

D. David Parras Armenteros

Dra. D^a. Macarena Espinilla Estévez

Agradecimientos

Quiero expresar mi más sincero agradecimiento a todas aquellas personas que han contribuido a que este proyecto sea una realidad, con especial mención a:

“A Iván Palomares Carrascosa por haber dejado un código muy bueno para su futura ampliación y por prestarme su ayuda cada vez que la he necesitado.

Mis padres y familia, porque gracias a su comprensión y confianza me ha permitido llegar hasta aquí.

Mi tutora Macarena Espinilla, por su paciencia al comprender la dificultad que tiene llevar un proyecto desde la distancia”.

GRACIAS A TODOS

Índice general

Introducción	15
1.1 Introducción al Proyecto	17
1.2 Propósito	20
1.3 Objetivos	21
1.4 Estructura y Planificación del Proyecto.....	21
Sistemas de Recomendación	25
2.1. Introducción	27
2.1.1. Esquema Básico de Funcionamiento y Elementos de un SR.....	29
2.1.2. Utilidad de los Sistemas de Recomendación.....	30
2.1.3. Datos en los SR.....	31
2.1.3.1. Realimentación en los SR	31
2.1.3.2. Realimentación Implícita.....	31
2.1.3.3. Realimentación Explícita	31
2.1.3.4. Datos Reales vs. Datos Sintetizados	32
2.1.3.5. Análisis Online vs. Análisis Offline.....	32
2.2. Clasificación de los Sistemas de Recomendación	32
2.2.1. Sistemas de Recomendación Basados en Contenido.....	33
2.2.2. Sistemas de Recomendación Colaborativos	35
2.2.3. Sistemas de Recomendación Basados en Conocimiento.....	36
2.2.4. Sistemas de Recomendación Híbridos	38
2.3. Ejemplos Reales de Sistemas de Recomendación en Internet	39
2.3.1. YouTube.....	39
2.3.2. Amazon.....	40
2.3.3. IMDb Recommendation Center	42
2.4. Sistemas de Recomendación Colaborativos	43
2.4.1. Introducción y Orígenes	43
2.4.2. Fases y Ejemplo de Funcionamiento de los SR Colaborativos	44
2.4.3. Algoritmos de Filtrado Colaborativo	46
2.4.4. Medidas y Técnicas Empleadas en Algoritmos de Filtrado Colaborativo.....	49

2.4.4.1. Notación	49
2.4.4.2. Algoritmo K-nn	50
2.4.4.3. Medidas de Similitud.....	51
2.4.4.4. Medidas de Predicción	53
2.4.5. Sistemas de Recomendación Colaborativos en Internet	54
2.4.5.1. Zagat.....	54
2.4.5.2. MovieLens	55
2.4.6. Problemas de los Sistemas de Recomendación Colaborativos.....	56
2.4.6.1 Arreglo del problema de Cold Start	58
Radios Colaborativas y Redes Sociales en Internet.....	61
3.1. Introducción a la difusión de contenidos multimedia online	63
3.2. Música en la Web: Streaming	63
3.3. Ejemplos de radios colaborativas en Internet.....	66
3.3.1. Last.fm.....	66
3.3.2. Pandora	68
3.4. Difusión de las redes sociales.....	69
3.5. Ejemplos de redes sociales en Internet	70
3.5.1. Facebook	70
3.5.2. MySpace	71
3.5.3. Twitter	72
3.6. Licencias de libre distribución de contenidos.	73
3.6.1. Copyleft	74
3.6.2. Creative Commons	75
3.6.2.1. Creative Commons en España.....	76
Desarrollo del Proyecto.....	81
4.1. Descripción del Proyecto.....	83
4.2. Especificación de Requerimientos	86
4.2.1. Requerimientos funcionales	87
4.2.2. Requerimientos No Funcionales	88
4.2.3. Requerimientos de la Interfaz.....	91
4.3. Análisis del Sistema	93
4.3.1. Modelo de Casos de Uso	93

4.3.2. Escenarios.....	114
4.4. Diseño del sistema	126
4.4.1. Diseño de los datos	127
4.4.1.1. Esquema conceptual	128
4.4.1.2. Esquema Conceptual Modificado	130
4.4.1.3. Tablas de la aplicación.....	132
4.4.2. Diseño de la Interfaz.....	139
4.4.2.1. Guía de Estilo.....	140
4.4.2.2. Metáforas	142
4.4.2.3. Pantallas de la aplicación	146
4.4.2.4. Caminos de navegación.....	156
4.5. Implementación	162
4.5.1. Tipo de arquitectura de la aplicación.....	162
4.5.2. Lenguajes de programación utilizados.....	163
4.5.3. Herramientas de desarrollo	165
Conclusiones	173
5.1. Conclusiones.....	175
 ANEXO A	
Manual de instalación del Servidor	179
A.1. Instalación de AppServ.....	181
A.2. Alojamiento del contenido de la aplicación en el servidor	188
A.3. Importar la base de datos y activar Drupal.....	189
 ANEXO B	
Manual de Usuario	193
B.1. Validación de un usuario.....	196
B.2. Utilización de las opciones de Menú Radio (usuario registrado)	200
B.3. Menú Personalizado de usuario (usuario registrado).....	217
B.4. Menú Mi Música (parte específica para usuario de tipo músico)	222
B.5. Menú Administrar Radio (parte específica para usuario de tipo administrador radio)	230

ANEXO C

Código fuente del Algoritmo de filtrado colaborativo	239
C.1. Contenido del fichero Main.java.....	242
C.2. Contenido del fichero Similitud.java	246
C.3. Contenido del fichero Predicción.java	248
Bibliografía y Referencias.....	251

CAPÍTULO 1

Introducción

1.1 Introducción al Proyecto

El uso, cada vez mayor, de los múltiples servicios que ofrece Internet por parte de los usuarios es de lo más variado, y está propiciando la existencia de sitios Web que sirven de soporte para muchos de estos servicios, no sólo en el ámbito científico, académico o empresarial, sino también para el ocio y disfrute del usuario.

La **radio por Internet** es actualmente uno de los mayores atractivos de ocio y tiempo libre para los internautas en general, debido a su facilidad de uso y su alto grado de accesibilidad. Se fundamenta en el **“Webcasting”** [16], es decir, la difusión a través de Internet de contenido multimedia, en este caso de audio. Para ello, se utiliza la tecnología conocida como **“Streaming”** [17], que consiste en brindar al usuario la posibilidad de reproducir contenidos multimedia directamente y de forma paralela en el navegador Web, sin que este tenga que descargar dicho contenido en su ordenador.

En este ámbito, uno de los software que más repercusión ha tenido ha sido **Spotify** [29], un revolucionario programa, lanzado a finales del 2008, que sirve para escuchar canciones vía streaming y adquisición de temas musicales. Además, permite la búsqueda de temas musicales por artista o álbum y la creación de listas de reproducción de los propios usuarios de la aplicación. La tecnología streaming está cambiando el concepto de música, ya que, según palabras de Daniel Ek, (fundador de Spotify), *“A la gente le da igual tener o descargarse canciones, lo que ellos realmente quieren es acceder a ellos”*.

Por otro lado, en el ámbito de la radio por Internet cabe destacar la reciente irrupción de radios personalizadas colaborativas, que ayudan al usuario a encontrar nueva música de su agrado, basándose en sus preferencias, es decir, estudiando las características de las canciones musicales que ha escuchado.

Algunos ejemplos de radios colaborativas los podemos encontrar en:

-**Pandora** (www.pandora.com): Mediante una interfaz muy conseguida [13], ayuda al usuario a generar una lista de artistas y canciones de su agrado desde el

momento en que escucha su primera canción, basándose en las similitudes entre objetos (canciones).

-**Last.fm** (www.lastfm.es): Considerada como una red social a gran escala, construye a partir de estadísticas de otros usuarios registrados perfiles sobre los gustos musicales que se adecúan al usuario. Su servicio es de código abierto y se basa en un algoritmo de filtrado colaborativo [3].

El exitoso funcionamiento de las citadas radios online radica en el uso de lo que formalmente se conoce como Sistemas de Recomendación Colaborativos.

Un **Sistema de Recomendación Colaborativo** puede definirse como:

Aquel sistema en el que las recomendaciones se hacen atendiendo a las similitudes entre usuarios, se trata de recomendar objetos del interés de otros usuarios que sabemos que tienen gustos e intereses similares.

Debido al éxito y difusión de las radios online personalizadas, en el año 2009 se presentó en la Universidad de Jaén el proyecto fin de carrera titulado: “**RADIO ONLINE basado en un motor de filtrado colaborativo**”, realizado por el alumno Iván Palomares Carrascosa [28]. Dicho proyecto desarrolla una radio colaborativa personalizada, mediante un algoritmo de filtrado colaborativo basado en ítem y provista de una interfaz Web. La radio desarrollada proporciona al usuario recomendaciones de música en forma de lista de reproducción, basada en las preferencias musicales del mismo, y permitiendo en todo momento puntuar las canciones, todas ellas bajo licencia Creative Commons [18]. Sobre la mencionada radio subyace una base de datos de 197 canciones.

La radio online personalizada ha sido un éxito. No obstante, presenta ciertas carencias y deficiencias que son descritas a continuación:

- Bajo la radio online personalizada subyace una pequeña base de datos. Dicho hecho origina que los usuarios habituales de la radio escuchen en un corto plazo de tiempo todas las canciones y debido al poco dinamismo de la radio el usuario deja de utilizarla.

- La radio online personalizada implementa un algoritmo de filtrado colaborativo que recomienda canciones en base a las puntuaciones dadas por los usuarios. Actualmente, la radio no ofrece la posibilidad de ejecutar el algoritmo de filtrado colaborativo vía Web, para realizar la actualización hay que realizar un proceso arduo y complejo, que puede consultarse en [31].
- Es una evidencia el actual auge de las **redes sociales en Internet**, estas son estructuras sociales compuestas de grupos de personas, las cuales están conectadas por uno o varios tipos de relaciones, tales como amistad, parentesco, intereses comunes o que comparten conocimientos [9]. Actualmente la mayoría de portales en internet cuentan con conexiones directas con los sistemas 2.0. Sin ir más lejos, la propia página de la Universidad de Jaén cuenta con conexiones con los sitios más populares, Youtube, Facebook o Twitter son algunos ejemplos, por otro lado se encuentra Spotify, que en su última versión se volvió también “social”, ya que se apoyaba en la red social Facebook. Por tanto, la radio online personalizada debe seguir avanzando e incorporar conexiones directas a este tipo de sistemas. De modo que los usuarios de la radio online personalizada puedan compartir sus gustos musicales a través de la conexión directa de la radio con las redes sociales con mayor auge.

El propósito del presente proyecto fin de carrera es solventar las carencias comentadas anteriormente y añadir nuevas funcionalidades que doten a la radio de valores añadidos con el fin de incrementar su calidad y los servicios que ofrece. Para ello proponemos:

- Aumentar la base de datos que subyace bajo la radio online personalizada. Para ello, se permitirá que cualquier músico o grupo musical se registre en la radio y pueda incorporar sus canciones a la base de datos. De este modo, el músico o grupo musical pueda promocionar sus canciones, a la vez que los oyentes de la radio disfrutan de ella, siempre que se acerque a sus gustos musicales. Es importante notar que las nuevas canciones que se incorporen

deberán poseer licencia Creative Commons, para mantener la política de la radio actual. Al incluir en la base de datos canciones nuevas que no poseen puntuaciones se nos origina el problema de arranque frío o **cold start** (el sistema de recomendación no puede extraer inferencias para los usuarios o temas sobre los que aún no ha reunido suficiente información). Por tanto, en el presente proyecto, propondremos un método para salvar dicho problema.

- Respecto a la tarea ardua y compleja de ejecución del algoritmo de filtrado colaborativo, en el presente proyecto se desarrollará un método más fácil y sencillo de ejecución. De modo que la ejecución del algoritmo se realice con un solo clic desde el sistema Web o mediante una tarea programada periódicamente.
- Como se ha comentado anteriormente, es importante la conexión directa de la radio online personalizada con redes sociales para que los usuarios compartan sus gustos musicales, por tanto una de las propuestas de este proyecto será desarrollar dicha conexión.
- Por último, pero no por ello menos importante, intentaremos mejorar la interfaz.

1.2 Propósito

Mejora y ampliación de funcionalidades de la radio online personalizada OL-Radio, con el fin de conseguir una mayor funcionalidad y poder potenciar su uso.

1.3 Objetivos

1. Búsqueda y revisión bibliográfica.
2. Incorporación de funcionalidad para la gestión de un grupo de música o músico de forma que pueda permitir el alta de estos, modificación de sus datos, crear nuevos álbumes, subir sus propias canciones y gestionar el tipo de licencia Creative Commons.
3. Desarrollar un método para solventar el problema del Cold Start presente en los algoritmos de filtrado colaborativo.
4. Creación de un nuevo usuario que permita la gestión de la radio online personalizada. Dicha gestión implica la revisión de nuevas canciones, gestión de usuarios, canciones y grupos musicales y, por último, la actualización del algoritmo de filtrado colaborativo.
5. Conectar la radio online personalizada con las redes sociales de mayor auge, de modo que los usuarios puedan compartir sus gustos musicales.
6. Realización de nuevos requerimientos para facilitar el uso a los usuarios como: listas con las canciones más escuchadas o con las canciones más puntuadas.
7. Dotar al sistema de una interfaz amigable y sobre todo atractiva, asimilándola a las páginas actuales de radios por internet y redes sociales.
8. Realización del manual de usuario tanto de oyente, usuario músico y usuario administrador.

1.4 Estructura y Planificación del Proyecto

A continuación, haremos una breve introducción a los diferentes capítulos en los que se estructura este proyecto y los contenidos expuestos en los mismos. Como hemos visto, este primer capítulo supone una introducción general al proyecto que se

ha realizado, con una justificación de su realización, la definición del propósito, los aspectos de la ampliación de la funcionalidad y los objetivos que persigue.

En el capítulo 2, abordamos una visión general de los Sistemas de Recomendación ya que el proyecto actual sigue contando con el mismo sistema de recomendación que en el anterior proyecto fue un éxito. En primer lugar, daremos la definición de Sistema de Recomendación, así como la justificación de su utilización. A continuación estudiaremos la estructura y elementos básicos de los Sistemas de Recomendación, y veremos una clasificación de Sistemas de Recomendación, atendiendo a su funcionamiento, ventajas e inconvenientes. Esta visión de los Sistemas de Recomendación irá acompañada de algunos ejemplos reales de sistemas de colaboración existentes en Internet. Por último, describiremos el problema del Cold Start y propondremos un método para solventar dicho problema en nuestro sistema.

El capítulo 3 está dedicado al concepto de radio colaborativa y red social en Internet. Haremos un breve repaso a las tecnologías de difusión de contenidos multimedia en Internet de *Streaming*, nos centraremos en la difusión de contenido musical, en las radios colaborativas y redes sociales, conociendo además algunos ejemplos reales de ellas. También revisaremos en este capítulo las licencias de distribución de contenidos, ya que es un importante factor a tener en cuenta por los usuarios que compartirán su música en OL-Radio.

El capítulo 4 supone el eje principal de la presente memoria, al ser el de mayor extensión y estar dedicado al proceso completo de desarrollo de este proyecto. Como proyecto software que es, haremos en primer lugar un breve repaso a las diferentes etapas de la Ingeniería de Software, para acto seguido aplicarlas en el desarrollo de nuestro sistema. Así, definiremos los requerimientos funcionales y no funcionales para el sistema nos centraremos en los nuevos y los que ya existían y han sido modificados, abordaremos la etapa de análisis, incluyendo el modelo de casos de uso en el sistema y el análisis de la base de datos. Acto seguido, pasaremos a la etapa de diseño, haciendo especial hincapié en el diseño de la nueva interfaz del sitio Web y las pruebas de usabilidad llevadas a cabo. Por último, repasaremos el proceso seguido para la etapa de implementación de la radio.

Una vez expuesto el desarrollo del proyecto, llegamos al quinto y último capítulo, dedicado a las conclusiones generales derivadas del desarrollo del mismo.

La sección final de esta memoria contiene los anexos dedicados a la instalación y configuración del servidor, el manual de usuario de la aplicación OL-RADIO para los tres usuarios: oyente, músico y administrador de la radio, y el código fuente del algoritmo de filtrado.

CAPÍTULO 2

Sistemas de Recomendación

2.1. Introducción

El actual auge de las Tecnologías de la Información, germen de lo que conocemos como Sociedad de la Información y su evolución hacia la Sociedad del Conocimiento, han propiciado que las personas dispongamos cada vez de más información para realizar nuestros acometidos. Esto es en cierto modo una ventaja, pero también nos encontramos con frecuencia el problema de sobrecarga de información, lo cual puede llegar a dificultar seriamente la tarea de escoger la información más adecuada a nuestras necesidades.

Es en Internet donde esta situación se hace presente a menudo, debido al cada vez mayor número de sitios dedicados a múltiples propósitos que ofrecen una considerable colección de información. El usuario necesita algún tipo de “ayuda” para elegir aquel contenido que sea de su interés. Así, en los últimos años los servicios de las citadas aplicaciones Web se han ido centrando en personalizar sus productos y/o servicios, de manera que consigan satisfacer las necesidades de cada usuario concreto.

Pensemos por ejemplo en una tienda online. El repertorio de productos que ofrecen dichas tiendas suele ser muy amplio y variado, lo cual provoca un gran número de elecciones posibles para los clientes y, por consiguiente, un aumento de la información que el cliente debe procesar para tomar una decisión. Esto puede traducirse, en muchos casos, en una sobrecarga de información, lo que hace que el usuario tenga una sensación de saturación que podría, en el peor caso, provocar que el usuario renuncie a realizar la compra al desistir de realizar una elección costosa en tiempo. En cambio, si el sistema de la tienda online fuese capaz de recomendar, entre el amplio conjunto de productos, aquellos que se estimen de mayor interés para cada usuario, la elección será mucho más sencilla para este.

Es aquí donde interviene una herramienta ampliamente utilizada, de forma satisfactoria, para resolver este inconveniente: se trata de los Sistemas de Recomendación.

Una primera y sencilla definición de Sistema de Recomendación es la siguiente:

Cualquier sistema que provee recomendaciones, predicciones u opiniones, normalmente en forma de conjunto de ítems, con el fin de ayudar al usuario a descubrir y evaluar ítems.

En adelante, utilizaremos el término *ítem* para referirnos a cualquier objeto, servicio o producto existente en un Sistema de Recomendación. Otra definición, empleada por algunos autores para los Sistemas de Recomendación, es la siguiente:

Aquellos sistemas que utilizan las opiniones de los usuarios de una comunidad para ayudar a los usuarios de la misma a encontrar contenidos de su gusto entre un conjunto sobrecargado de elecciones posibles.

A partir de estas definiciones, no nos resulta muy difícil concluir que los Sistemas de Recomendación actúan de un modo muy similar a como lo hace la inteligencia humana, al basarse en actos tan frecuentes en la vida cotidiana como: pedir asesoramiento (recomendaciones) a expertos en un dominio particular, guiarse por individuos con gustos y preferencias similares a las del usuario (lo que se conoce como Sistema de Recomendación Colaborativo), o elegir objetos con cualidades similares a las de otros objetos que ya han sido positivamente valorados por el usuario con anterioridad (Sistema de Recomendación basado en contenido).

Actualmente existen varios tipos de Sistemas de Recomendación [1], entre los que destacan los Sistemas de Recomendación Colaborativos, basados en contenido, basados en conocimiento e híbridos. Veremos con mayor profundidad las características de cada uno de ellos en un epígrafe posterior.

Tanto los Sistemas de Recomendación Colaborativos como los basados en contenido presentan el inconveniente de requerir una gran cantidad de información, tanto de usuarios del mismo como de los ítems que lo componen, para realizar recomendaciones de calidad y, por tanto, funcionar correctamente. Para resolver este inconveniente aparecieron otros tipos de Sistemas de Recomendación capaces de realizar recomendaciones de calidad sin necesidad de una gran cantidad de información: se trata de los Sistemas de Recomendación basados en conocimiento. Por último, de cara a mejorar aún más los resultados de los Sistemas de Recomendación,

han aparecido en los últimos años los Sistemas de Recomendación híbridos, en los que se sintetizan las ventajas de dos o más de los anteriores tipos de Sistemas de Recomendación, para así lograr un resultado mejor al que cada uno de ellos lograría por separado.

A continuación, estudiaremos brevemente la estructura básica de funcionamiento de los Sistemas de Recomendación (que en adelante también nombraremos de forma abreviada como SR), los principales elementos que los componen, y la utilidad de los mismos.

2.1.1. Esquema Básico de Funcionamiento y Elementos de un SR

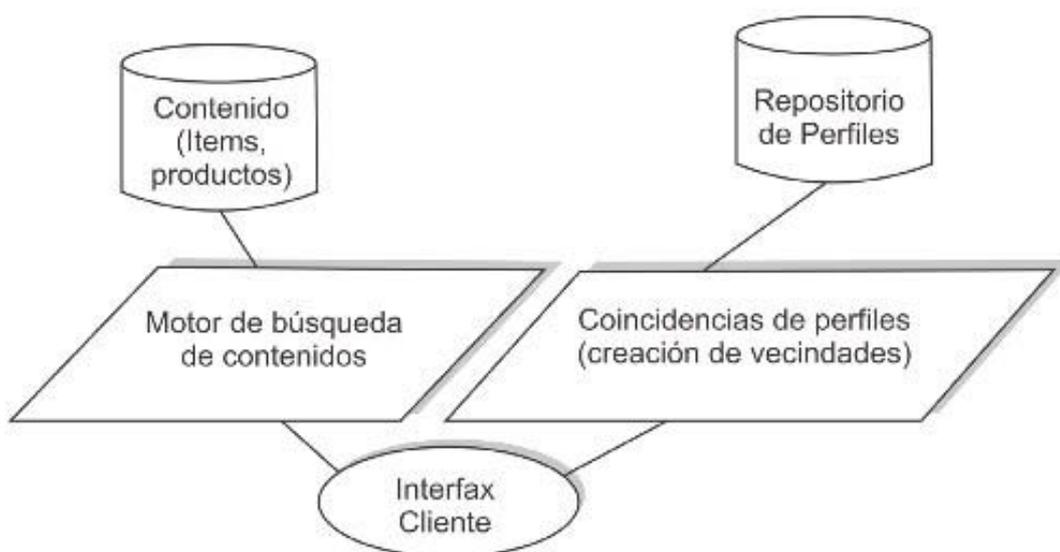


Figura 2.1: Esquema básico de funcionamiento de un SR

En la figura 2.1 se puede observar un esquema básico de un SR en el que se puede distinguir los siguientes elementos:

- **Base de datos:** La calidad de los datos almacenados en nuestra base de datos jugará un papel crucial a la hora de realizar recomendaciones con mayor o menor calidad.

- **Perfiles de Usuario:** Un usuario va “dándole forma” a su perfil personal a medida que utiliza el sistema. El perfil refleja los gustos/preferencias del usuario, fundamentales a la hora de discriminar objetos durante la recomendación.

- **Predicciones:** La predicción es el elemento clave en el esquema básico de todo SR. La predicción se basa en el perfil del usuario y en la información disponible en la base de datos con la que contamos.

2.1.2. Utilidad de los Sistemas de Recomendación

Los Sistemas de Recomendación han sido utilizados con gran éxito en aplicaciones de marketing personalizado en Internet, dado que reducen el tiempo invertido en realizar búsquedas de productos y hacen las búsquedas más efectivas, lo cual conlleva una mayor satisfacción a los clientes. Además, en los últimos tiempos están cobrando importancia los Sistemas de Recomendación en portales de difusión de contenidos multimedia, tales como música o vídeos. En ellos, el usuario recibe recomendaciones de canciones o vídeos que se prevé que serán de su agrado.

Para conseguir estos objetivos, todos los SR llevan a cabo dos tareas básicas:

- **Predecir:** Los Sistemas de Recomendación realizan predicciones de un conjunto de objetos, productos o servicios en los cuales podría estar interesado un usuario o cliente concreto.

- **Recomendar los N-mejores objetos:** Se trata de determinar los N objetos en los que un usuario o cliente estará más interesado.

En el siguiente epígrafe trataremos algunas cuestiones más específicas de los Sistemas de Recomendación, tales como la política de tratamiento de los datos que utilizan, además de los mecanismos de realimentación empleados.

2.1.3. Datos en los SR

Es importante tomar algunas decisiones a la hora de desarrollar un Sistema de Recomendación, tales como la forma en que recibirá información nueva, el tipo de datos a utilizar o la forma en que dichos datos se analizarán.

2.1.3.1. Realimentación en los SR

Un Sistema de Recomendación no debe ser una entidad estática, sino que la calidad de sus recomendaciones ha de evolucionar con el tiempo, en base a la experiencia y nueva información adquirida. Esto se consigue mediante los mecanismos de realimentación entre el sistema y los usuarios. Existen dos mecanismos de realimentación: la realimentación implícita y la realimentación explícita.

2.1.3.2. Realimentación Implícita

Un mecanismo de realimentación implícita es aquel que proporciona al SR información sobre las preferencias de los usuarios sin que estos sean conscientes de ello. Estas realimentaciones no se hacen de forma directa, sino mediante algunas medidas como pueden ser: el tiempo de visualización del objeto, el número de consultas del mismo, etc.

Presenta el problema de que depende demasiado del contexto y es bastante hipotética, ya que se hacen suposiciones (a partir de las mencionadas medidas) sobre los gustos del usuario que no necesariamente tienen por qué ser ciertas.

2.1.3.3. Realimentación Explícita

La realimentación explícita se basa en la acción directa y deliberada del usuario para indicar aquellos objetos del sistema que le interesan. Esta acción se puede conseguir mediante votaciones numéricas o, simplemente, indicando si el objeto es o no del agrado del usuario. Este tipo de realimentación también presenta problemas, como son la voluntariedad del cliente o el tiempo invertido en ello.

2.1.3.4. Datos Reales vs. Datos Sintetizados

Otra cuestión interesante es la de elegir un conjunto de datos reales (recopilados de usuarios reales sobre objetos reales) o un conjunto de datos sintetizados (sin ninguna base real, específicamente creados para el Sistema de Recomendación). Estos últimos son más fáciles de obtener, ya que evitamos tener que realizar encuestas u otros métodos de recopilación de información real, aunque sólo se utilizan en las primeras fases del desarrollo del sistema, para posteriormente ser sustituidos por los datos reales una vez que se haya recopilado la información suficiente.

2.1.3.5. Análisis Online vs. Análisis Offline

Es importante decidir si vamos a trabajar sobre los datos de manera online u offline. En el análisis offline se emplea una técnica o algoritmo de filtrado para hacer predicciones sobre el conjunto de datos, evaluando los resultados de dichas predicciones mediante una o varias métricas de error. Este tipo de análisis presenta la ventaja de ser rápido y económico, pero tiene a la vez dos inconvenientes importantes: el problema de la escasez de datos y el problema de obtener como único resultado la bondad de la predicción.

Por el contrario, el análisis online permite obtener más resultados, entre los que destacan la actuación de los usuarios participantes, la satisfacción de los mismos, etc. Sin embargo, resulta ser más lento y costoso que el análisis offline.

2.2. Clasificación de los Sistemas de Recomendación

Una vez sentadas las bases de los Sistemas de Recomendación, es el momento de revisar los diferentes tipos existentes de los mismos. Los Sistemas de Recomendación pueden ser implementados utilizando diversas técnicas [1], por lo que su clasificación dependerá de su funcionamiento para calcular recomendaciones. La clasificación más habitual es la siguiente:

2.2.1. Sistemas de Recomendación Basados en Contenido

Un Sistema de Recomendación basado en contenido es aquel que está basado en las características de los objetos, es decir, las recomendaciones se llevan a cabo basándose únicamente en un perfil creado a partir del análisis del contenido de los objetos que el usuario ha evaluado con anterioridad.

En otras palabras, estos sistemas extraen características de los objetos y las comparan con el perfil del usuario para predecir las preferencias de los usuarios sobre tales objetos. La idea es recomendar objetos similares en su contenido a objetos que ya sabemos que son del agrado del usuario en cuestión, es decir, los que forman parte de su perfil.

El filtrado basado en contenido era el tipo de Sistema de Recomendación más extendido antes de que se produjese la explosión del filtrado colaborativo, ya que los sistemas del primer tipo tienen un claro problema de sobre-especialización.

El funcionamiento de los sistemas de colaboración basados en contenido (Figura 2.2) comprende dos grandes etapas: analizar las descripciones de los productos valorados por los usuarios y predecir qué productos le pueden gustar.

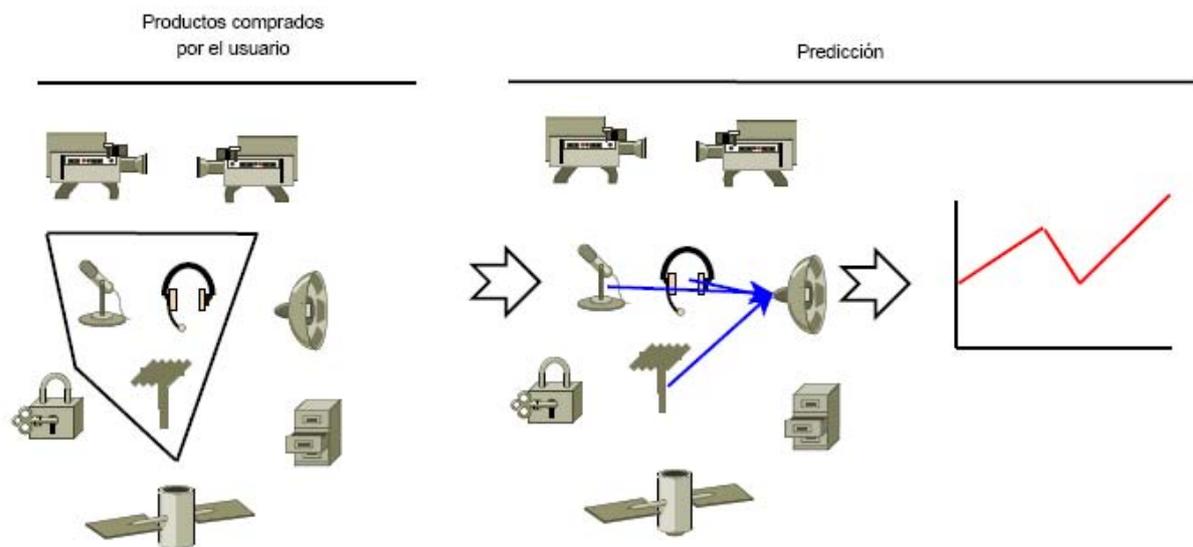


Figura 2.2: Esquema de funcionamiento de un Sistema de Recomendación basado en contenido.

Este funcionamiento dependerá en gran medida del tipo de información descriptiva que se utilice en el perfil del usuario. Esta información se divide en:

- *Conjunto de características:* Asociado a cada producto tenemos un conjunto de características que lo describen [2]. Por ejemplo, kilometraje o consumo medio si estamos hablando de vehículos.

- *Información textual sobre el producto:* Se trata de un documento que describe al producto. Se diferencia del conjunto de características principalmente en que la información no está estructurada.

Muchos Sistemas de Recomendación basados en contenido trabajan únicamente en base a información del conjunto de características de los objetos. En tal caso, el funcionamiento es similar al mostrado en la figura 2.3.

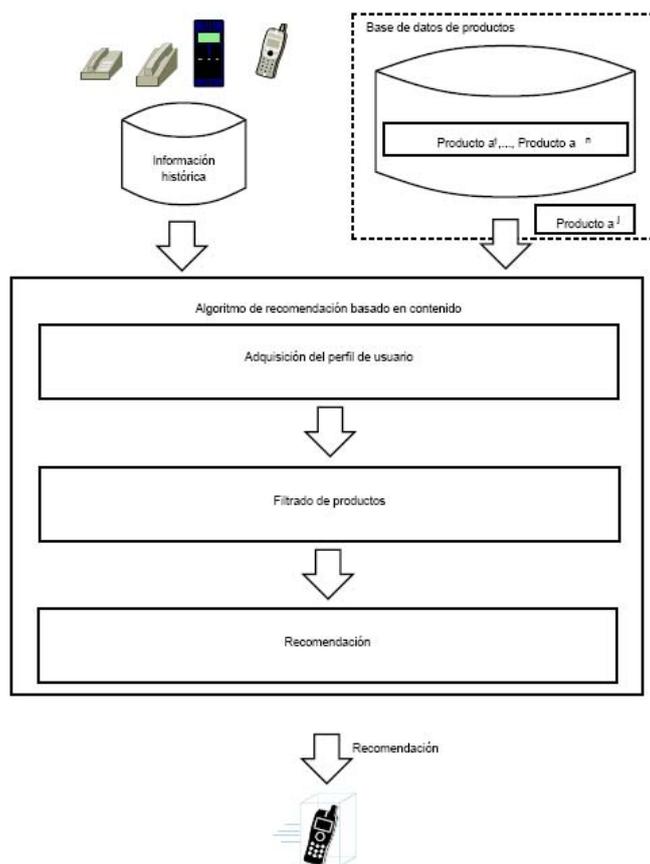


Figura 2.3: Funcionamiento de un SR basado en contenido.

Los Sistemas de Recomendación basados en contenido resultan de gran utilidad para la recomendación de documentos y páginas Web, donde el cálculo de similitud entre objetos da unos resultados bastante satisfactorios.

Este tipo de Sistema de Recomendación presenta dos grandes inconvenientes:

- **Sobre-especialización:** Se da como consecuencia de reducir las recomendaciones a unos contenidos muy similares, sin dar importancia a la posible arbitrariedad de los gustos e intereses del usuario.

- **Sólo se puede conocer información parcial de un objeto,** normalmente información textual, dejando así de lado información contextual, visual o semántica, la cual puede resultar muy útil para realizar las recomendaciones, pero al mismo tiempo es la más difícil de obtener. Como consecuencia de ello, pueden perderse conexiones no inmediatas entre objetos similares.

Ante estos problemas, se han propuesto múltiples soluciones, entre las que cabe destacar las siguientes:

- Incorporación de cierta aleatoriedad en las búsquedas.
- Indexación semántica latente (LSI) de la información textual.
- Medidas de similitud basadas en ontologías.

No obstante, la mejor solución a dichos problemas ha resultado ser la existencia de una buena realimentación entre el sistema y sus usuarios.

2.2.2. Sistemas de Recomendación Colaborativos

Se trata de los Sistemas de Recomendación más extendidos y consolidados en el mercado actual. Los Sistemas de Recomendación basados en un filtrado colaborativo son aquellos que realizan recomendaciones basándose en los términos de similitud entre usuarios, esto es, recomiendan objetos que son del agrado de otros usuarios de intereses similares.

Para la realización de un buen Sistema de Recomendación Colaborativo, que ofrezca recomendaciones de calidad, es fundamental emplear un buen algoritmo de filtrado colaborativo. Estos algoritmos se pueden clasificar en dos categorías: los algoritmos basados en memoria o usuario y los algoritmos basados en ítem. La mayor ventaja de las técnicas colaborativas es que son totalmente independientes de la representación interna de los objetos que se pueden recomendar.

Sin embargo, a medida que se ha extendido su utilización, se han detectado problemas como son: la escasez, la escalabilidad y el problema del ítem nuevo. Por ello, han llevado a cabo multitud de estudios y experimentos orientados a minimizar el efecto de estos problemas.

Este modelo de Sistemas de Recomendación constituye el principal eje en torno al cual versa el presente proyecto, por lo que veremos un estudio más detallado de los mismos en un epígrafe posterior.

2.2.3. Sistemas de Recomendación Basados en Conocimiento

Los Sistemas de Recomendación basados en conocimiento realizan un razonamiento sobre las necesidades de un usuario y sus preferencias, haciendo inferencias para sugerir recomendaciones [23]. Este tipo de Sistemas de Recomendación emplea el conocimiento que tiene sobre usuarios y productos para generar recomendaciones sobre qué productos satisfarán las necesidades o gustos del usuario.

En la siguiente figura podemos ver un ejemplo del funcionamiento de los Sistemas de Recomendación basados en conocimiento, utilizando un razonamiento basado en casos para generar las recomendaciones:

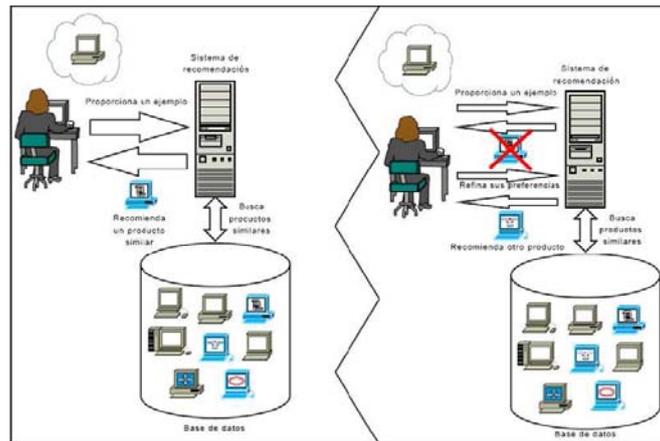


Figura 2.4: Funcionamiento de los Sistemas de Recomendación basados en conocimiento.

Veamos las fases que sigue el esquema de funcionamiento:

1. *Fase de obtención de un ejemplo de las necesidades del usuario:* El usuario proporciona al sistema un ejemplo que represente sus necesidades.
2. *Fase de búsqueda de productos que cumplen las necesidades del usuario:* A partir de la descripción del ejemplo proporcionado por el usuario, el SR buscará otros productos similares a él y los devolverá como recomendaciones (Figura 2.4: Parte izquierda).
3. *Fase de refinamiento:* En algunas circunstancias puede ocurrir que los productos recomendados a partir del ejemplo no cubran con exactitud las necesidades del usuario. En estos casos, el usuario debe refinar las características del producto seleccionado como ejemplo, añadiendo y/o modificando nuevas características (Figura 2.4: Parte derecha).

La principal ventaja de este tipo de sistemas es que, a diferencia de otros Sistemas de Recomendación, los basados en conocimiento no necesitan grandes cantidades de información sobre los objetos evaluados (contenido) o sobre los perfiles de los usuarios (colaborativos). Sólo necesitan un conocimiento general sobre el conjunto de objetos y un conocimiento informal de las necesidades del usuario. Además, presentan muchos beneficios y unos resultados exitosos al ser hibridados con otro tipo de Sistemas de Recomendación.

Los SR basados en conocimiento presentan 2 grandes inconvenientes:

- La mayor parte del conocimiento que utilizan para realizar recomendaciones debe ser explícitamente proporcionado por el usuario, por lo que muchas veces requieren un esfuerzo humano considerable para que las heurísticas de inferencia empleadas realicen recomendaciones de calidad.

- Problemas en la fase de refinamiento: Es muy difícil que un usuario encuentre un ejemplo exacto de lo que necesita. Por ello, es bastante probable que éste deba refinar su perfil de usuario modificando o añadiendo algunas de las características del ejemplo dado.

2.2.4 Sistemas de Recomendación Híbridos

Como hemos mencionado, cada uno de los modelos de recomendación anteriormente vistos presenta, además de sus ventajas, algún tipo de problema. Por ello, surgió la idea de solventar esta situación aunando sus puntos fuertes e intentando minimizar sus inconvenientes, mediante la hibridación de al menos dos de los diferentes tipos de Sistemas de Recomendación. De esta forma nacieron los Sistemas de Recomendación híbridos.

Los sistemas híbridos entre los basados en contenido y los colaborativos, por ejemplo, guardan las preferencias del usuario y las combinan con los objetos más relevantes para realizar las recomendaciones.

Existen también los sistemas híbridos entre los colaborativos y los basados en conocimiento, los basados en contenido y en conocimiento, e incluso entre los colaborativos y las redes sociales.

2.3. Ejemplos Reales de Sistemas de Recomendación en Internet

A continuación, citaremos algunos Sistemas de Recomendación que pueden ser encontrados en conocidos portales de Internet.

2.3.1. YouTube

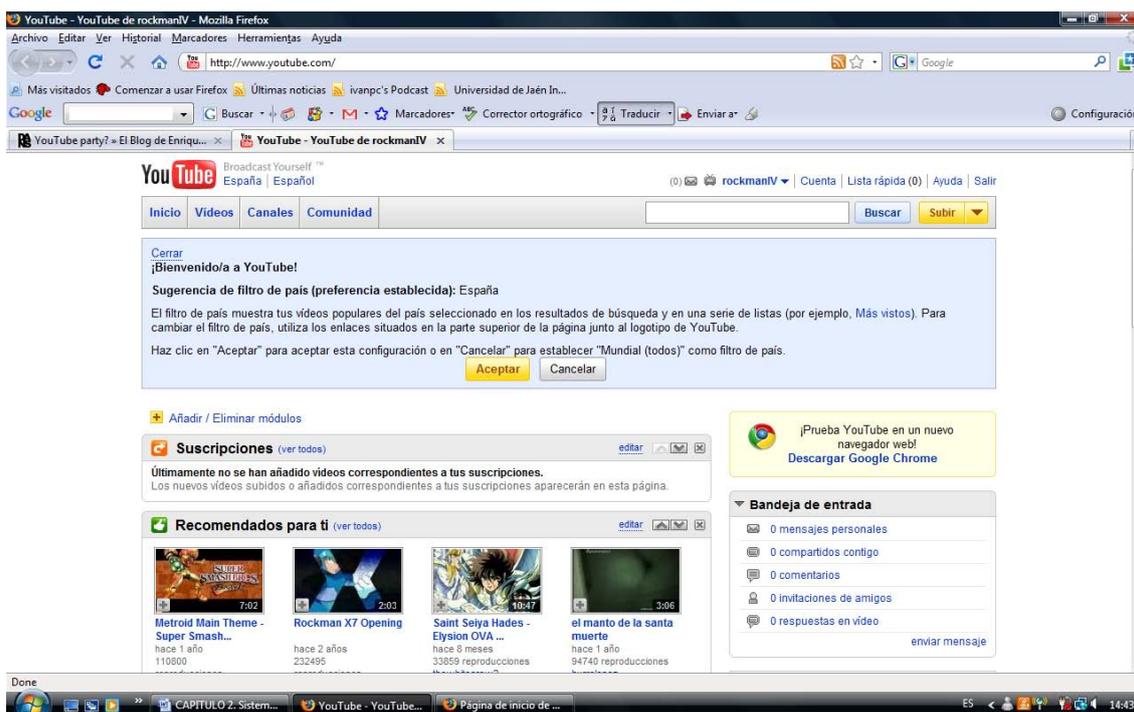


Figura 2.5: Portal de YouTube en Internet

El ya famosísimo portal de difusión de vídeos a través de Internet, Youtube, ha incorporado recientemente un Sistema de Recomendación basado en el contenido, en el que los usuarios registrados obtienen un conjunto de vídeos recomendados, similares a aquellos que suelen ver con mayor frecuencia o durante un porcentaje prolongado de la duración de estos. Dicho modelo de recomendación se plasma en una interfaz fácilmente editable y flexible.

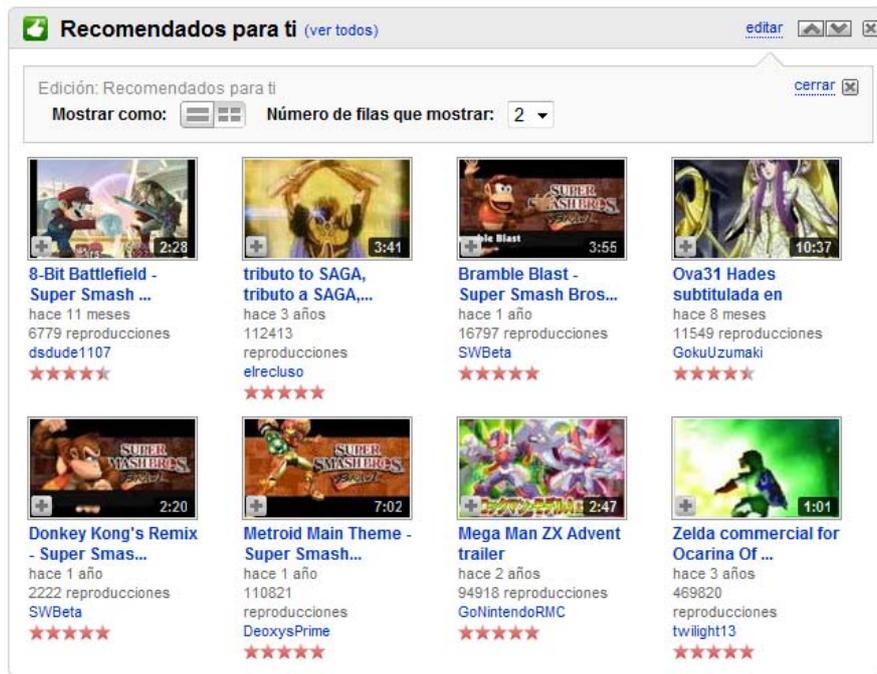


Figura 2.6: Interfaz de recomendaciones de YouTube

2.3.2. Amazon

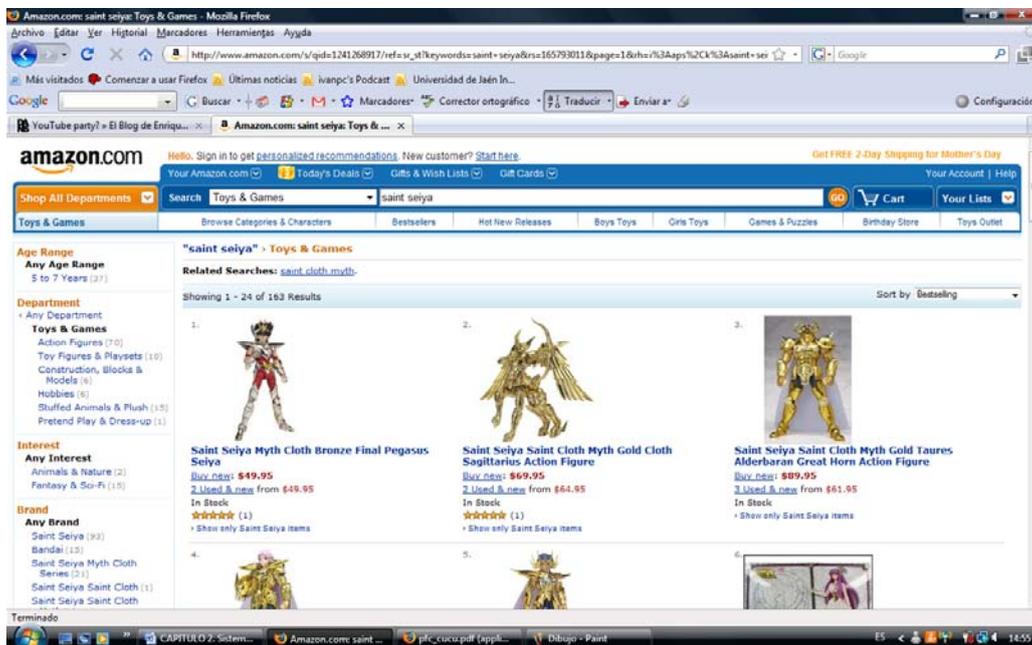
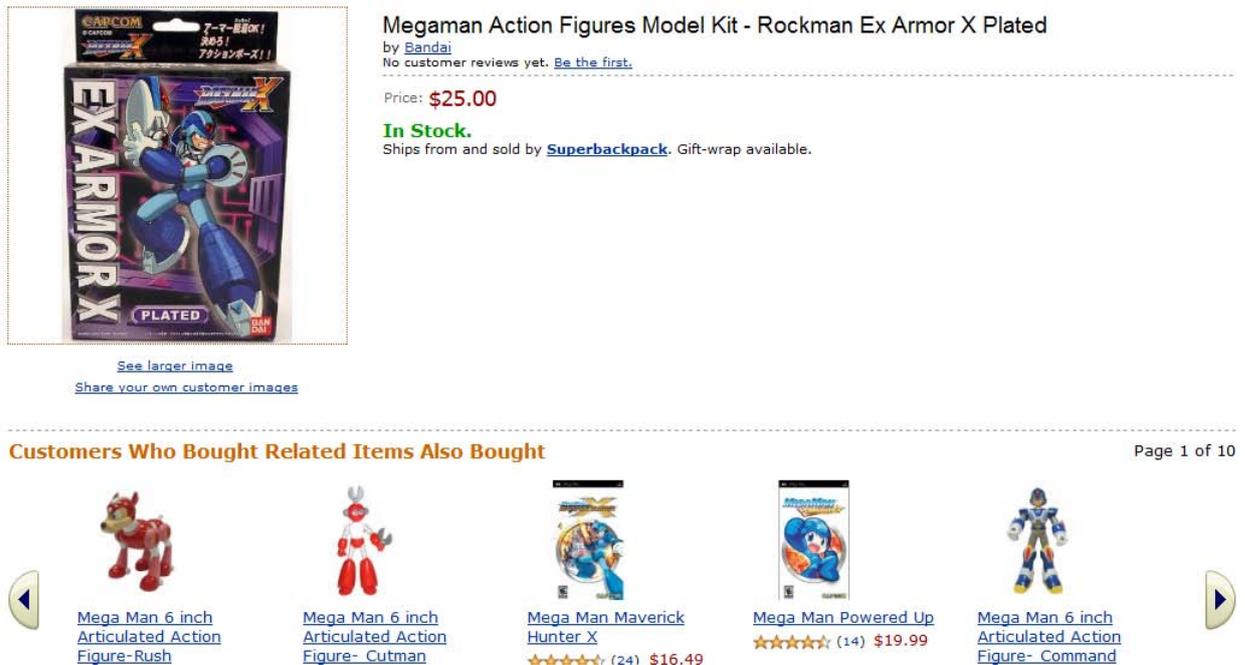


Figura 2.7: Portal de Amazon en Internet

La popular empresa norteamericana de comercio electrónico es ya todo un referente en el mundo de los Sistemas de Recomendación, con un paradigma que

combina las ventajas de los enfoques colaborativo y basado en contenido. El sistema guarda las preferencias del usuario para posteriormente combinarlas con objetos relevantes para realizar las recomendaciones. Dichas recomendaciones se muestran al usuario en el tradicional formato: “People who bought this ítem... also bought...”).



Megaman Action Figures Model Kit - Rockman Ex Armor X Plated
by [Bandai](#)
No customer reviews yet. [Be the first.](#)

Price: **\$25.00**

In Stock.
Ships from and sold by [Superbackpack](#). Gift-wrap available.

[See larger image](#)
[Share your own customer images](#)

Customers Who Bought Related Items Also Bought Page 1 of 10

- 
[Mega Man 6 inch Articulated Action Figure-Rush](#)
- 
[Mega Man 6 inch Articulated Action Figure- Cutman](#)
- 
[Mega Man Maverick Hunter X](#)
★★★★☆ (24) \$16.49
- 
[Mega Man Powered Up](#)
★★★★☆ (14) \$19.99
- 
[Mega Man 6 inch Articulated Action Figure- Command](#)

Figura 2.8: Interfaz de recomendación de Amazon

2.3.3. IMDb Recommendation Center

La Internet Movie DataBase (IMDb) es la mayor base de datos del mundo con información en línea sobre películas, directores, actores, series de televisión, etc. La IMDb cuenta con un motor de recomendaciones basado en contenido, llamado Recommendation Center.

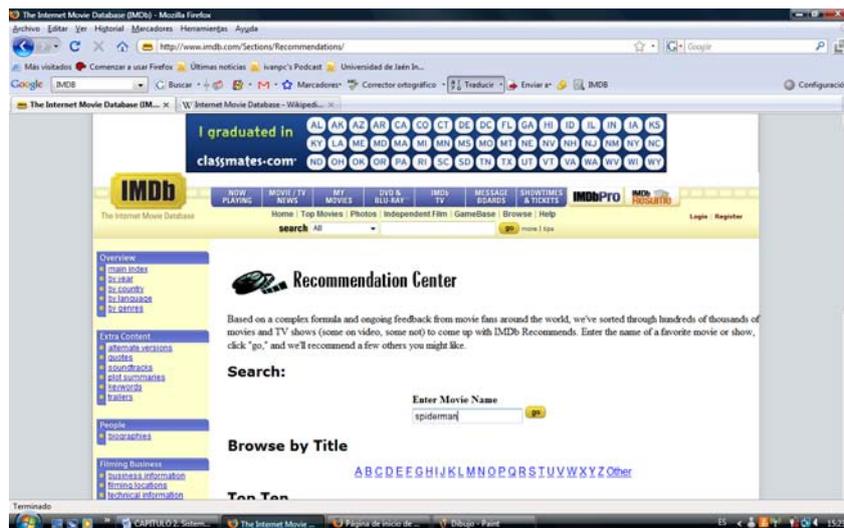


Figura 2.9: Interfaz del centro de recomendación de IMDb



Figura 2.10: Interfaz del centro de recomendación de IMDb

El usuario introduce el programa de televisión o película que le guste y el sistema le ofrece una lista con al menos diez recomendaciones (Figura 2.10). Además,

incorpora un mecanismo de realimentación en el que el usuario indica los elementos que no sean de su agrado entre aquellos que le han sido recomendados, para así ir recibiendo recomendaciones nuevas, más depuradas. Pese a ser muy famoso, no es un motor de recomendación demasiado bueno, por lo que no es el punto fuerte de esta compañía.

2.4. Sistemas de Recomendación Colaborativos

Debido a que este proyecto se sustenta sobre un algoritmo de filtrado colaborativo, a continuación vamos a hacer una revisión un poco más en detalle de este tipo de Sistemas de Recomendación [6].

2.4.1. Introducción y Orígenes

Los Sistemas de Recomendación Colaborativos son aquellos que realizan recomendaciones basándose únicamente en términos de similitud entre los usuarios. Es decir, combinan las valoraciones de los objetos, identifican los gustos comunes entre usuarios en base a dichas valoraciones y recomiendan así objetos que son del gusto de otros usuarios de gustos similares al usuario actual.

Las técnicas para desarrollar los primeros Sistemas de Recomendación de filtrado colaborativo estaban basadas en métodos provenientes de la minería de datos [2]. Para ello, se distinguía entre una fase de aprendizaje (offline) en la que se aprende el modelo, al igual que ocurre en la minería de datos, y una fase de recomendación (online) en la que se aplica el modelo obtenido de la fase anterior a un problema de la vida real, produciéndose así las recomendaciones para los usuarios del sistema. No obstante, actualmente este tipo de técnica no se suele utilizar, ya que debido a la interacción de los usuarios con el sistema es más conveniente emplear un paradigma de aprendizaje relajado (el modelo se construye y actualiza durante el funcionamiento del sistema).

La base teórica de los Sistemas de Recomendación es simple: se forman grupos de usuarios más cercanos, que serán los que mantengan unos perfiles similares, y a un usuario de un grupo se le recomiendan objetos que aún no ha experimentado, pero sí han experimentado y valorado positivamente otros usuarios de su “grupo”.

2.4.2. Fases y Ejemplo de Funcionamiento de los SR Colaborativos

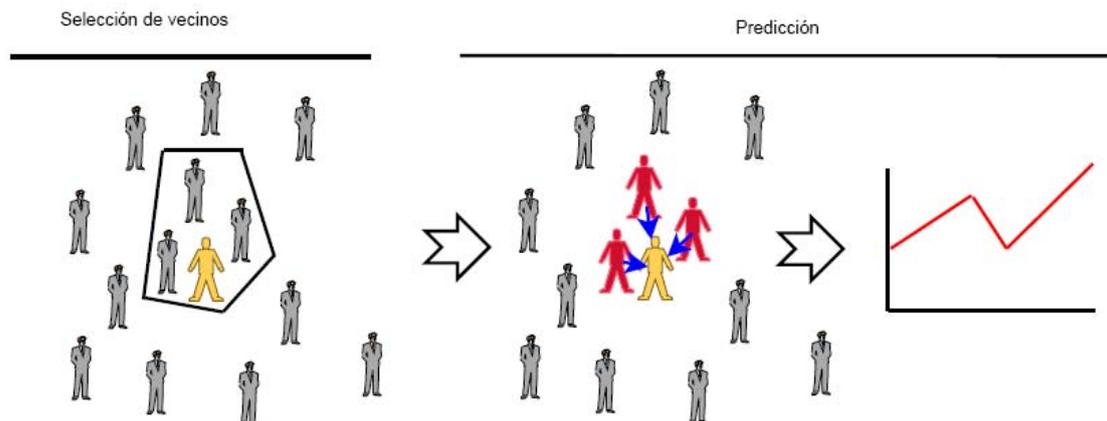


Figura 2.11: Funcionamiento de los Sistemas de Recomendación Colaborativos.

Shardanand y Maes [10] distinguieron tres etapas fundamentales en el funcionamiento de todo Sistema de Recomendación Colaborativo, que puede verse gráficamente en la Figura 2.11.

1. El sistema guarda un perfil de cada usuario, que consta de las evaluaciones de objetos conocidos por él y que pertenecen a la base de datos sobre la que se trabajará.
2. En base a estos perfiles, se mide el grado de similitud entre los usuarios del sistema y se crean grupos de usuarios con características similares.
3. El sistema utiliza la información obtenida en los dos pasos anteriores para calcular las predicciones (Figura 2.12). A cada usuario se le recomendarán objetos que no haya evaluado previamente y que hayan obtenido los mayores valores para dicha predicción.

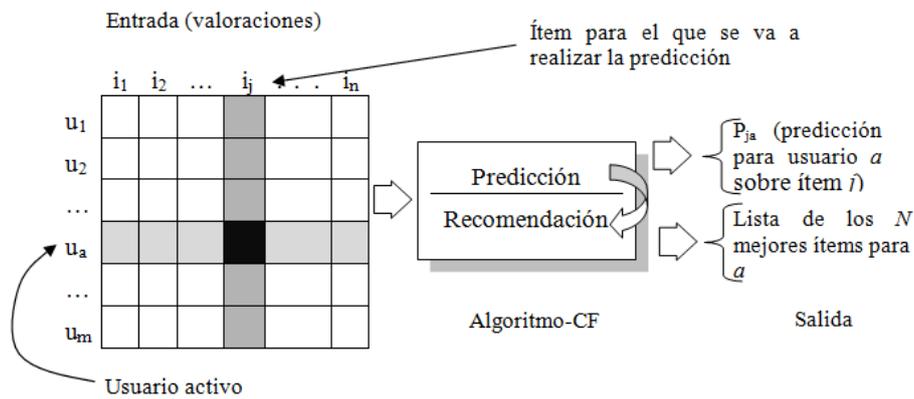


Figura 2.12: Proceso de cálculo de predicciones en SR Colaborativos.

Por tanto, estos Sistemas de Recomendación no toman en consideración el contenido y las características de los productos que recomiendan, sino que sean del gusto de usuarios con un perfil semejante al del usuario que demanda el servicio.

Para entender mejor el funcionamiento de estos sistemas, utilizaremos el siguiente ejemplo que muestra una relación de usuarios y salones de celebraciones de la provincia:

	Juan	Luis	Manuel	Pedro	Antonio
Sierra Mágina	+	-	+	+	-
Puerta Cerro	+	+	+	+	+
Mistral	-	+	-	+	+
California	+	-	+	-	-
Ruta del Sol	-	+	-	+	?

Figura 2.13: Ejemplo de funcionamiento de SR Colaborativos.

El símbolo “+” significa que al usuario le ha gustado la descripción del lugar, el símbolo “-” indica que al usuario no le ha gustado, y el símbolo “?” representa la recomendación que queremos calcular. Queremos saber, pues, la valoración que podría tener Antonio sobre los salones Ruta del Sol. Para hacer esta predicción, buscaremos usuarios que tengan un patrón de valoraciones (perfil) similares a Antonio. Podemos ver que Luis tiene unas valoraciones muy parecidas a las de Antonio, por lo que podríamos suponer que, dado que a Luis le gustan los salones Ruta del Sol, a Antonio

también le gustarán. Esta solución no sería del todo correcta, ya que al basarse solo en un usuario con gustos parecidos puede ofrecer recomendaciones equivocadas.

Lo ideal es buscar el grado de correlación entre Antonio y cada uno de los usuarios del sistema, y usar dicho grado para realizar una media ponderada de sus valoraciones. Una forma de obtener el grado de correlación entre un usuario “x” y otro usuario “y” es mediante el coeficiente de correlación de Pearson, donde $r(x,y)$ representa la valoración del usuario “x” sobre el producto “y”. Así, la correlación entre los usuarios “x” e “y” se puede obtener de la siguiente forma:

$$r(x,y) = \frac{\sum_{d \in \text{Productos}} (R_{x,d} - \bar{R}_x) (R_{y,d} - \bar{R}_y)}{\sqrt{\sum_{d \in \text{Productos}} (R_{x,d} - \bar{R}_x)^2 (R_{y,d} - \bar{R}_y)^2}}$$

Donde \bar{R}_x es el valor medio de las valoraciones hechas por el usuario “x”.

Siguiendo con el ejemplo, sabemos que la correlación entre Antonio y Luis es 1, entre Antonio y Pedro es 0.577, entre Antonio y Juan es 0.577, y entre Antonio y Manuel es -0.577. Si calculamos media ponderada de la valoración de cada usuario sobre los salones Ruta del Sol por su correlación con Antonio, obtendremos un valor de 0.682.

Un algoritmo de filtrado colaborativo predeciría, pues, que a Antonio le gustarían los salones Ruta del Sol. En el siguiente apartado veremos una clasificación de los algoritmos de filtrado colaborativo.

2.4.3. Algoritmos de Filtrado Colaborativo

Para desarrollar un buen Sistema de Recomendación Colaborativo, es de vital importancia elegir un buen algoritmo de filtrado colaborativo [5]. Existen distintas posibilidades a la hora implementar dicho algoritmo. Veamos con un poco de detalle los diferentes tipos de algoritmos de filtrado [8].

- Algoritmos basados en memoria o basados en usuario.

Estos algoritmos realizan las recomendaciones basándose en la base de datos completa, teniendo en cuenta todos los ítems previamente evaluados por el usuario. El funcionamiento de los algoritmos es el siguiente: se utilizan técnicas estadísticas para determinar un conjunto de vecinos al usuario objetivo (Figura 2.14) y, posteriormente, se aplican algoritmos que combinan preferencias de esta vecindad para realizar las predicciones y recomendaciones.

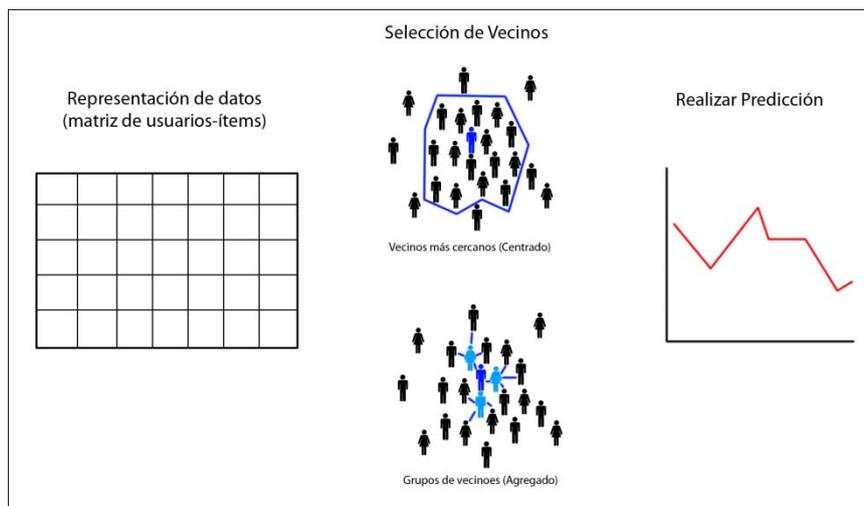


Figura 2.14: Las principales partes de un SR Colaborativo basado en memoria

Pese a ser bastante populares y exitosos en la práctica, suelen sufrir especialmente los problemas de escalabilidad y escasez, que analizaremos posteriormente. Se hizo necesario, pues, el desarrollo de otro tipo de algoritmos de filtrado colaborativo.

- Algoritmos basados en modelos o basados en ítem

Estos algoritmos se centran en desarrollar un modelo de las puntuaciones de los usuarios sobre los ítems, para proporcionar las recomendaciones. Se utiliza para ello una aproximación probabilística que calcula el valor esperado de una predicción del usuario a partir de sus puntuaciones sobre otros ítems.

Dado que este es el tipo de algoritmo sobre el que se centrará el presente proyecto, pasamos a continuación a describir distintas técnicas para obtener el modelo de recomendación.

- *Redes bayesianas*: Esta técnica consiste en obtener el modelo a partir de un conjunto de entrenamiento con un árbol de decisión, donde los nodos y ramas representan información de los usuarios. Suele resultar útil cuando el conocimiento sobre el perfil de usuario cambia de forma lenta.
- *Modelos basados en reglas*: El procedimiento es similar al de la técnica anterior, con la diferencia de que el modelo obtenido tiene la forma de un conjunto de reglas del tipo “Antecedente → Consecuente”.
- *Técnicas de agrupamiento (clustering)*: Hacen grupos de usuarios, denominados *clusters*, con gustos similares. Las predicciones para un individuo se realizan mediante la agregación de opiniones de otros usuarios del mismo grupo. La tarea de creación de los *clusters* conlleva un gran esfuerzo, pero una vez creados se obtiene un rendimiento bueno, ya que el grupo a considerar para realizar las recomendaciones queda considerablemente reducido.
- *Modelos basados en vecino más cercano*: Estos algoritmos proporcionan recomendaciones desarrollando en primer lugar un modelo, utilizando cualquiera de las tres técnicas anteriormente descritas, de las puntuaciones de los usuarios sobre los ítems.

Estos algoritmos miran en el conjunto de ítems evaluados por el usuario activo para calcular cómo de parecidas son estas puntuaciones al ítem activo, con el fin de realizar una predicción para el mismo. Para realizar las recomendaciones se realizan las siguientes tareas:

1. Exploración del conjunto de ítems que el usuario ha valorado.
2. Cálculo de la similitud de los ítems anteriores con respecto al ítem o producto del cual queremos predecir la puntuación que el usuario le daría.
3. Selección de los k ítems más cercanos (Knn).

4. Cálculo de la predicción del usuario objetivo sobre el ítem dado como la media ponderada de las valoraciones del usuario hacia los k ítems más similares.

Dentro de los Sistemas de Recomendación Colaborativos basados en ítem, emplearemos este último modelo para el algoritmo de filtrado de este proyecto.

2.4.4. Medidas y Técnicas Empleadas en Algoritmos de Filtrado Colaborativo

Una vez estudiada la clasificación de algoritmos de filtrado colaborativo, pasamos a definir algunas medidas empleadas para realizar las recomendaciones en este tipo de sistemas.

El primer paso para la realización de dichas recomendaciones consiste en formar grupos con los usuarios o ítems más similares entre sí. Para ello se emplearán medidas de similitud y un algoritmo de clasificación K- nn [7]. Acto seguido, se empleará una técnica de predicción para estimar la valoración del usuario sobre ciertos ítems.

2.4.4.1. Notación

Antes de mostrar formalmente la formulación del problema, es conveniente dejar clara la notación que se va a emplear para no producir confusión al lector:

-Un usuario es aquel elemento representado por $u_i \in U = \{u_1, u_2, \dots, u_n\}$

-Un ítem será aquel elemento representado por $i_i \in I = \{i_1, i_2, \dots, i_m\}$

-La similitud entre dos ítems i_j e i_k vendrá dada por $s(i_j, i_k)$

-Una evaluación de un usuario u_i sobre un ítem i_j vendrá dada por r_{u_i, i_j}

-Una predicción de un usuario u_i sobre el ítem i_j se representará como p_{u_i, i_j}

2.4.4.2. Algoritmo K-nn

Un paso crucial en la realización de un Sistema de Recomendación Colaborativo de calidad es la formación de grupos de usuarios (si es un SR Colaborativo basado en memoria) o de ítems (si es basado en modelos, como ocurre en este proyecto) de características similares. Esta actividad forma parte de lo que conocemos como problemas de clasificación, y existen diversas técnicas, llamadas clasificadores, para resolver dicho problema. Uno de los clasificadores más difundidos es el algoritmo K-nn, que se utilizará en este proyecto para formar los grupos de ítems más similares para cada uno de los ítems de la base de datos.

Veamos con mayor detalle en qué consiste el algoritmo K-nn:

a) Funcionamiento

Siendo i el objeto a clasificar, debemos seleccionar los k objetos con $K = \{i_1, \dots, i_k\}$ tal que no existe ningún ejemplo i' fuera de K con $d(i, i') < d(i, i_j), j = 1, \dots, k$

Una vez encontrados los k vecinos, se puede proceder a la clasificación de dos formas posibles:

-Voto por la mayoría: Se clasifica el nuevo objeto según la clase predominante en los objetos de K .

-Voto compensado por la distancia: Se clasifica el objeto en base a su distancia ponderada con el resto de objetos de K .

b) Descripción del algoritmo

1. Se separan los datos en dos conjuntos disjuntos: entrenamiento (E) y test (T).
2. Aparece un nuevo objeto i_a .
3. Se obtienen los k objetos del conjunto E más cercanos a i_a

4. Se clasifica el objeto i_a de una de las dos formas antes mencionadas.¹

c) Principales características

-Es un algoritmo robusto frente al ruido cuando k es moderado ($k > 1$).

-Es muy eficaz cuando el número de clases posibles es alto y cuando los datos son heterogéneos o difusos.

-Tiene un orden de complejidad de $O(dn^2)$, siendo $O(d)$ la complejidad de la distancia de métrica empleada.

-El hecho de no utilizar modelos sino la base de datos al completo provoca que sea ineficiente en memoria.

-Sirve tanto para realizar clasificación como para predicción numérica.

En nuestro proyecto se empleará el algoritmo K-nn para seleccionar los k ítems más similares a cada uno de los ítems que componen nuestra base de datos.

2.4.4.3. Medidas de Similitud

Para establecer la similitud entre objetos debemos definir una medida que nos permita evaluar el grado de parecido entre unos y otros.

Es importante resaltar que, en este proyecto, para calcular la similitud entre dos ítems x e y se tendrán en cuenta únicamente a aquellos usuarios que hayan evaluado a ambos ítems, no siendo tomados en consideración el resto.

¹ En este proyecto nos limitamos a utilizar el algoritmo K-nn para seleccionar los k vecinos más cercanos a cada uno de los objetos, por lo que no necesitamos implementar este último paso.

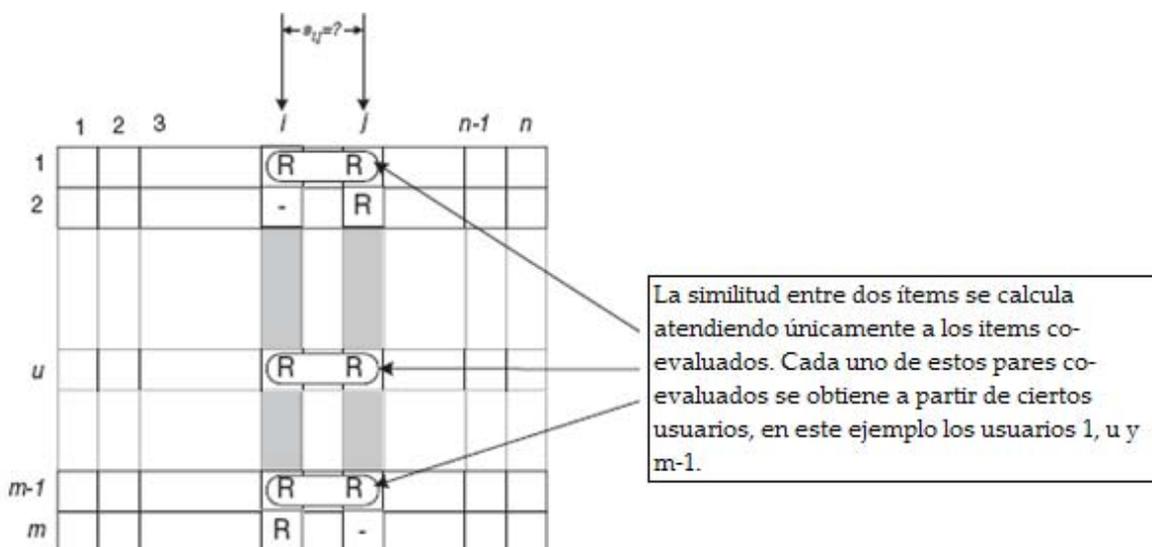


Figura 2.15: Cálculo de similitud basada solamente en ítems co-evaluados.

Existen en la literatura multitud de medidas de similitud [4][10], de entre todas, nosotros vamos a revisar dos de las más utilizadas.

1. Coeficiente de Correlación de Pearson

Este coeficiente es un índice que mide la relación lineal entre dos variables cuantitativas, siendo independiente de la escala de medida de dichas variables y dando un resultado dentro del intervalo [-1,1]. Se calcula mediante la siguiente expresión:

$$s(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}}$$

Siendo \bar{x} e \bar{y} la media de las valoraciones obtenidas para los ítems x e y .

2. Coeficiente Coseno

Este método supone que dos ítems x e y vienen representados por vectores en el espacio, por lo que la similitud entre ellos vendrá dada por el coseno del ángulo que forman. La expresión para su cálculo es la siguiente:

$$s(x, y) = \frac{\sum_{i=1}^n x_i y_i}{\sqrt{\sum_{i=1}^n (x_i)^2 \sum_{i=1}^n (y_i)^2}}$$

Siendo x_i el valor del objeto x para el usuario i , y_i el valor del objeto y para el usuario i y n el número de usuarios que han evaluado tanto x como y .

En este proyecto se ha elegido el coeficiente coseno como medida de similitud para la implementación del algoritmo de filtrado colaborativo.

2.4.4.4. Medidas de Predicción

Tras haber calculado el conjunto de vecinos para cada ítem, hemos de combinar las valoraciones de dicho conjunto para realizar la predicción del usuario sobre dicho ítem. Elegir la técnica adecuada para realizar la predicción supone el paso más crucial del filtrado colaborativo.

La selección de un algoritmo de predicción u otro depende de la naturaleza del conjunto de datos, ya que cada algoritmo se ajusta mejor a un conjunto de datos específico. En nuestro caso, utilizaremos la técnica llamada **weighted sum**. Este método calcula la predicción de un ítem i por parte de un usuario u_a como la suma de las evaluaciones del usuario u_a sobre ítems similares a i . Cada una de estas evaluaciones viene ponderada por la correspondiente similitud $s(i, j)$ entre los ítems i y j . Su expresión viene dada por:

$$p(u_a, i_a) = \frac{\sum_{h=1}^k s(i_a, i_h) * r_{u_a, i_h}}{\sum_{h=1}^k |s(i_a, i_h)|}$$

Siendo k los k ítems más similares a i_a . Esta técnica intenta captar cómo el usuario activo evalúa a ítems similares al que quiere predecir. Para asegurarnos de que la predicción entra del rango previamente definido, es necesario ponderar estas evaluaciones con la similitud.

Ya hemos sentado las bases teóricas sobre las que se apoya nuestro proyecto para implementar nuestro algoritmo de filtrado colaborativo, pasamos a mostrar algunos ejemplos reales de SR Colaborativos existentes en la Web.

2.4.5. Sistemas de Recomendación Colaborativos en Internet

Tras estudiar los fundamentos teóricos de los Sistemas de Recomendación Colaborativos, pasamos a ver algunos ejemplos reales de sitios Web que emplean este tipo de sistemas.

2.4.5.1. Zagat



Figura 2.16: Portal de Zagat.com

Zagat Survey es una empresa norteamericana fundada en 1979, dedicada a la edición de todo tipo de guías de restaurantes, tiendas, hoteles y clubes de diferentes ciudades de los Estados Unidos y Canadá. El usuario registrado puede valorar hasta 30 aspectos diferentes del local referido, además de introducir breves comentarios dando su opinión o experiencia en el mismo. A partir de estas valoraciones, los responsables de la empresa asignan sus puntuaciones en sus guías anuales, que servirán para realizar las recomendaciones a los usuarios a través de la Web.

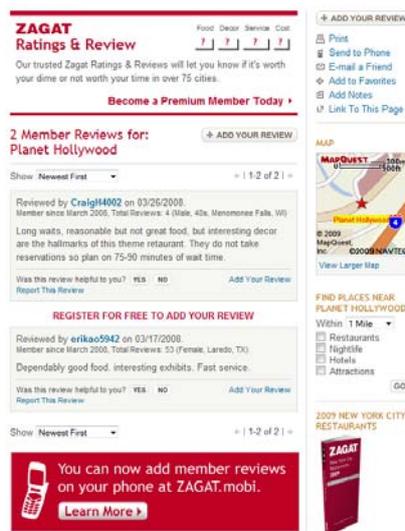


Figura 2.17: Opiniones de usuarios en Zagat.com

2.4.5.2. MovieLens

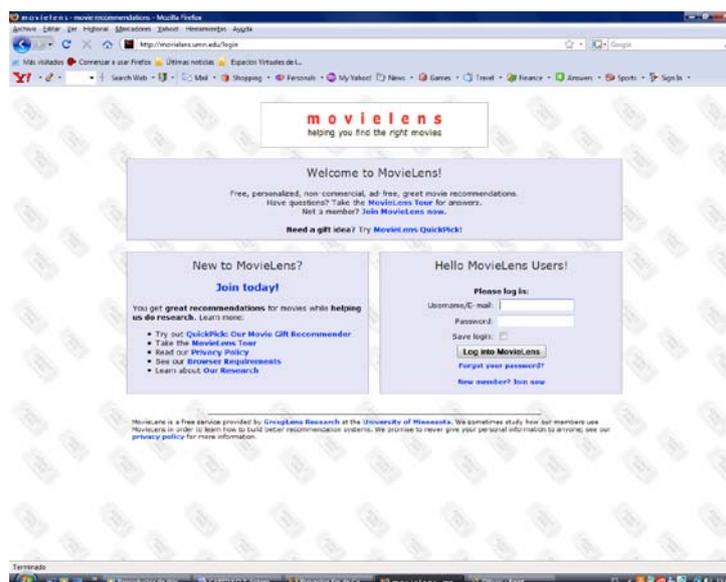


Figura 2.18: Portal de MovieLens

MovieLens (Figura 2.18) es un Sistema de Recomendación de películas online, basado en un motor de filtrado colaborativo, desarrollado por el grupo investigador GroupLens Research (<http://www.grouplens.org>) de la Universidad de Minnesota. Su principal función como Sistema de Recomendación Colaborativo es la de recopilar puntuaciones de los usuarios sobre películas, y emplear esta información para agrupar a los usuarios de gustos similares. Así, basándose en las puntuaciones de los usuarios del grupo al que pertenece un usuario activo, se intenta predecir para dicho usuario su valoración sobre películas aún no evaluadas por él.

Aunque la mayoría de datos de puntuaciones son privados, MovieLens mantiene como públicas dos muestras de 100.000 y un millón de puntuaciones, respectivamente.

Y para terminar, en el siguiente apartado citaremos los principales problemas que han surgido de la utilización de Sistemas de Recomendación Colaborativos.

2.4.6. Problemas de los Sistemas de Recomendación Colaborativos

Como hemos mencionado con anterioridad, los Sistemas de Recomendación Colaborativos presentan una serie de problemas que se han ido poniendo de relieve a medida que su utilización se ha extendido: escasez, escalabilidad y arranque frío o Cold Start. Para tener un mejor conocimiento de estos, los describimos brevemente a continuación:

- Escasez

Dada su naturaleza, los Sistemas de Recomendación Colaborativos requieren una gran cantidad de usuarios que realicen un igualmente grande número de puntuaciones sobre ítems similares, para poder así calcular los grupos de ítems cercanos y realizar recomendaciones de calidad. Si el número de usuarios registrados en el sistema es pequeño, o incluso si siendo elevado no han hecho suficientes puntuaciones, los cálculos de

vecindad, predicción y recomendación serán deficientes y darán como consecuencia recomendaciones de poca calidad.

- Escalabilidad

Para obtener la similitud entre usuarios, los Sistemas de Recomendación Colaborativos utilizan algoritmos de cálculo del vecino más cercano (Knn). El problema de estos algoritmos es su elevado coste computacional, que crecerá de forma lineal respecto al número de elementos existentes en la base de datos.

- Arranque frío o Cold Start

El problema de Cold Start lo podemos dividir en dos subproblemas:

- Problema del ítem nuevo

Este problema repercute sobre los dos elementos principales del Sistema de Recomendación, los usuarios y los ítems. En esencia, un ítem nuevo en el sistema apenas tendrá puntuaciones respecto a otros ítems ya existentes, por lo que no van a ser recomendados prácticamente nunca.

- Problema del Usuario Nuevo (ó early rater)

Con respecto a los usuarios, un usuario nuevo en el sistema habrá realizado pocas puntuaciones sobre los ítems existentes. Esto provoca que encuadrarlos en un grupo de vecinos adecuado sea una labor compleja, por lo que las recomendaciones que reciba serán pobres.

En los últimos tiempos se ha realizado una gran cantidad de experimentos, estudios e investigaciones de cara a reducir el impacto de estos problemas.

Para reducir el problema de la escasez se han utilizado técnicas de puntuaciones implícitas, correlación entre ítems y filtrado híbrido. Para tratar el problema de la escalabilidad se ha optado por una técnica muy empleada en el análisis

de datos [7]: la reducción de la dimensionalidad, además de aproximaciones basadas en modelos.

Tras haber estudiado los principales problemas de los Sistemas de Recomendación Colaborativos. En nuestro sistema nos centraremos en la solución del problema de Cold Start, en el siguiente apartado trataremos con detalle cómo afecta este problema en nuestro sistema y formularemos una posible solución.

2.4.6.1 Arreglo del problema de Cold Start

Como vimos anteriormente, el problema de Cold Start es considerado cuando un nuevo usuario o un nuevo ítem entran en el sistema.

Con respecto al problema de nuevos usuarios, en la mayoría de los casos se ataca este problema exigiéndoles a los nuevos usuarios que tomen postura sobre una serie de elementos. De modo que se pueda determinar un perfil inicial para poder realizarle una primera recomendación. Por tanto, para solucionar este problema en nuestro sistema se les exigirá a los nuevos usuarios que realicen algunas puntuaciones para así poder obtener su perfil.

Así, hemos presentado una solución al problema de Cold Start para un nuevo usuario.

Sobre el problema de nuevo ítem, en nuestro sistema se traducirá en que algunas canciones nunca serán recomendadas al no tener ninguna puntuación. Así, para solucionar dicho problema y conseguir que las canciones del gusto de los usuarios puedan ser recomendadas tengan la posibilidad de ser recomendadas por la radio a los usuarios se propone la siguiente técnica:

En primer lugar, se contabilizará las puntuaciones que posee cada canción. De las diez canciones que recomienda la radio, seguiremos utilizando ocho en base a las predicciones del algoritmo de filtrado colaborativo y las dos canciones restantes serán escogidas aleatoriamente pero de las cinco canciones

(del estilo musical favorito del usuario) que menos puntuaciones hayan recibido de toda la base de datos.

Resumiendo, pongamos el ejemplo de que ha entrado una nueva canción en la base de datos esta contará con cero puntuaciones y nunca será recomendada, así que cuando un usuario, que tenga como estilo musical favorito el mismo estilo de la canción, utilice la radio tendrá muchas posibilidades de escuchar esta canción al contar con cero puntuaciones. Así, podemos decir que con el método propuesto permite vencer el problema de Cold Start para nuevos ítems, ya que el usuario escuchará dos canciones que carecen de elevadas puntuaciones y permitirá que el usuario las puntué. De este modo, ayudaremos a que las nuevas canciones sean puntuadas y dichas puntuaciones sirvan para recomendarlas, siempre que se acerquen al gusto musical del usuario.

Debemos mencionar que en este caso es posible utilizar esta técnica, ya que si la canción recomendada al usuario no es adecuada, no supone un daño elevado, ya que el usuario puede pasar la canción o puntuarla negativamente. Para concluir, vemos que el daño asociado a una mala predicción no es muy elevado. Esto no podría llevarse a cabo por ejemplo en otros Sistemas de Recomendación de hoteles o restaurantes porque no sería adecuado que nos recomendará un hotel que no es de nuestro agrado.

CAPÍTULO 3

Radios Colaborativas y Redes Sociales en Internet

3.1. Introducción a la difusión de contenidos multimedia online

Hace algunos años, Internet no era más que un vago concepto para la mayoría de las personas. Hoy día, la Web tiene cientos de millones de usuarios habituales en todo el mundo, y la Web tal y como hoy la conocemos es muy diferente de lo que era hace apenas unos años. Los sitios Web que contienen únicamente texto e imágenes estáticas están siendo rápidamente reemplazados por un nuevo concepto de sitio Web [16], en el que podemos realizar acciones tan variadas como ver programas de televisión, hablar por video y audio-conferencias, escuchar clips de música, escuchar estaciones de radio en directo, e incluso asistir a una clase a distancia.

Así, el concepto de *Streaming multimedia* [17] está actualmente presente en la Web: nos referimos, por ejemplo, a emisoras de radio o televisión, retransmisión de conciertos, etc. Son cada vez más las compañías que proporcionan vídeo y audio a través de servidores multimedia distribuidos.

El gran impulso y aceptación que hasta el momento se le ha dado al contenido interactivo y dinámico de elementos multimedia transmitidos a través de la red está cambiando la forma en que las personas se comunican y acceden a la información. Este gran impulso es debido al gran desarrollo de las tecnologías de la información y la comunicación en los últimos años.

3.2. Música en la Web: Streaming

Streaming es un término utilizado para denominar la tarea de ver u oír un archivo multimedia (vídeo, audio...) de forma directa en una página Web, sin necesidad de descargar dicho archivo en el ordenador. Se podría describir como “hacer un clic y obtener”. En términos más específicos, puede definirse como sigue:

Una estrategia sobre demanda para la distribución de contenido multimedia a través de Internet.

Así, el Streaming de audio hace posible escuchar música a través de una página Web sin necesidad de descargar los archivos musicales a escuchar, ya que estos quedan almacenados en un buffer en el instante en que están siendo reproducidos.

Antes de que la tecnología Streaming apareciese en el año 1995, con el lanzamiento del sistema de transmisión de música online Real Audio 1.0, era necesario descargar el archivo de audio en el disco duro del usuario para la reproducción de música existente en Internet. Dado que estos archivos suelen tener un tamaño significativo, su descarga en el equipo podía suponer un proceso lento, especialmente en conexiones con límite de descarga no muy elevado.

Sin embargo, el Streaming permite descargar el archivo y reproducirlo al mismo tiempo, mientras aún está descargándose, consiguiendo así un tiempo de espera mínimo. Esta es la principal ventaja de esta tecnología, y lo que la ha hecho tan popular en los últimos años.

El funcionamiento del Streaming es el siguiente.

-En primer lugar, el ordenador del usuario (cliente) se conecta con el servidor (proveedor del sitio Web) y éste inicia el envío del fichero multimedia en cuestión.

-El cliente comienza a recibir el fichero y en ese preciso instante construye un buffer de memoria donde empieza a guardar la información.

-Cuando se ha llenado el buffer con una pequeña parte del archivo, el ordenador cliente empieza a reproducirlo, a la vez que continúa con la descarga. Normalmente el intervalo de tiempo necesario para ello es muy pequeño, casi nulo.

-El sistema está sincronizado para que el archivo se pueda reproducir mientras que el archivo se descarga, de modo que cuando el archivo acaba de descargarse el fichero ya lleva cierta cantidad de tiempo en reproducción.

-Si en algún momento la conexión sufre descensos de velocidad se utiliza la información que hay en el buffer, de modo que se pueda dar un margen temporal de tolerancia a dicho descenso en la conexión. Si la comunicación se cortase demasiado

tiempo, el buffer se vaciaría y la ejecución el archivo se cortarían también hasta que se restaurase la conexión.

Hoy en día podemos encontrar multitud de sitios en Internet en los que se saca partido a la tecnología Streaming. Un ejemplo de ello lo tenemos en las llamadas emisoras de radio o **radios online**, concepto en torno al cual gira el objetivo de este proyecto. Se trata aplicaciones Web en las que el usuario tiene acceso a una colección de archivos musicales para escuchar sin necesidad de descargarlos en su ordenador, además de disponer de un perfil personal, crear sus propias listas de reproducción con aquellas canciones que desee, etc. Muchas de estas radios online trabajan incluso sobre la emisión de audio en vivo, como ocurre con las emisoras de radio clásicas.

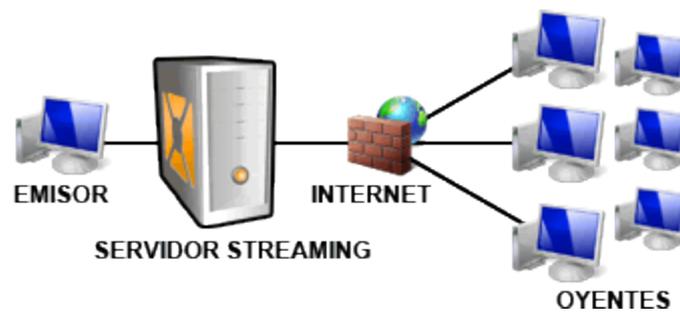


Figura 3.1: Ejemplo de estructura de audio-Streaming en directo.

En el siguiente epígrafe, veremos dos ejemplos de sitios radios online en Internet, provistas además de un Sistema de Recomendación Colaborativo para el usuario, según hemos estudiado en el capítulo 2.

3.3. Ejemplos de radios colaborativas en Internet

Este apartado presentamos dos radios online colaborativas ampliamente conocidas: Last.fm y Pandora.

3.3.1. Last.fm (www.lastfm.es)

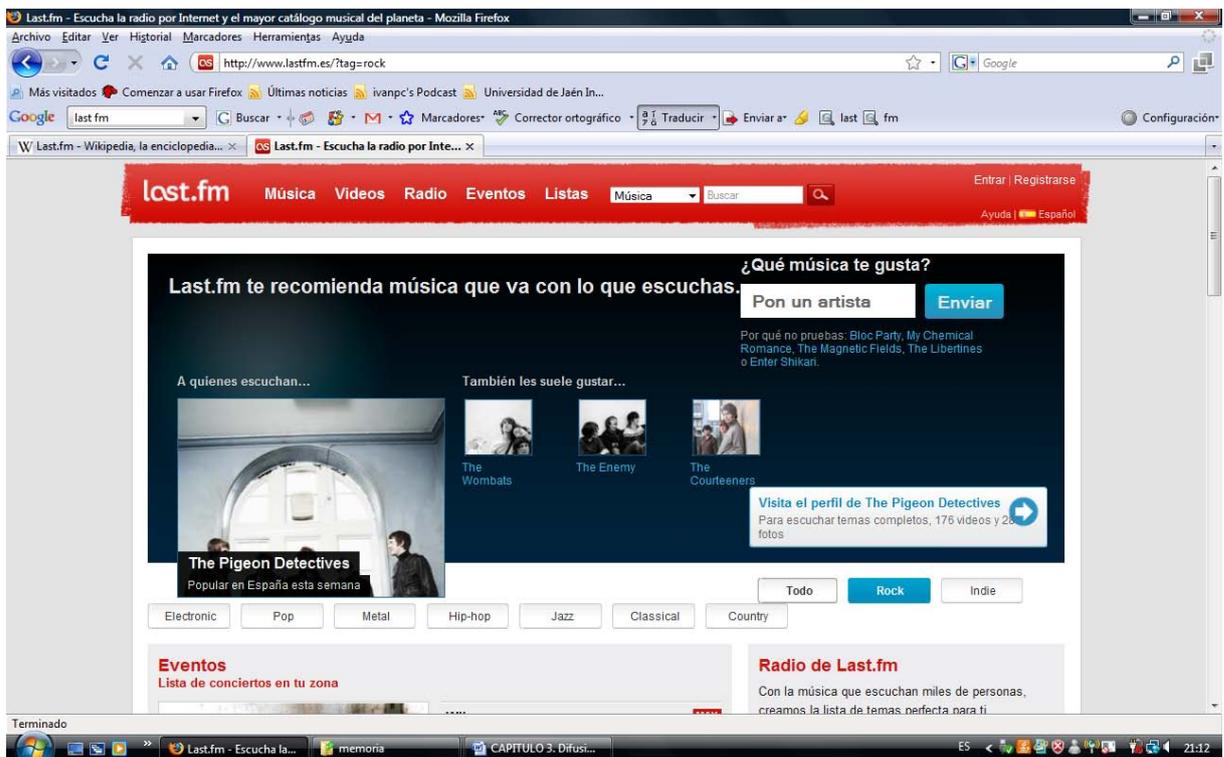


Figura 3.2: Portal de Last.fm

Last.fm es una famosísima red social, una radio online y además un Sistema de Recomendación Colaborativo de música, que se basa en los datos de los usuarios registrados para construir perfiles y estadísticas de gustos musicales, para realizar las recomendaciones. El servicio de que dispone es de código abierto. Se originó a partir de un proyecto de informática en la Universidad de Southampton (Reino Unido).

The screenshot shows the Last.fm website interface. At the top, there's a navigation bar with 'lost.fm' logo, 'Música', 'Videos', 'Radio', 'Eventos', and 'Listas'. A search bar is present with 'Música' selected. On the right, there are links for 'Entrar | Registrarse' and 'Ayuda | Español'.

The main content area is for the artist 'Muse'. On the left, there's a sidebar with navigation options: 'Artista', 'Biografía', 'Fotos', 'Videos', 'Álbumes', 'Temas', 'Eventos', 'Noticias', 'Listas', 'Artistas similares', 'Tags', 'Oyentes', 'Blog', and 'Grupos'.

The 'Biografía' section for 'Muse' states: '101,847,477 scrobblings (1,634,956 oyentes)'. It has '21,412 notas'. Below this, there are buttons for '+ Agregar a mi colección' and 'Compartir'. The biography text describes Muse as an alternative rock band from Teignmouth, Devon, England, with members Matthew Bellamy, Dominic Howard, and Chris Wolstenholme. It mentions their early names (Gothic Plage, Fixed Penalty, Rocket Baby Dolls) and their success.

Below the biography, there are 'Tags' (alternative rock, rock, alternative, progressive rock, indie) and a section for 'Artistas similares' (similar artists) including Placebo, Radiohead, Franz, Kasabian, The Killers, Kaiser Chiefs, and Coldplay.

On the right side, there's a music player for 'Muse - Time Is Running Out'. Below the player is a list of 'Temas más escuchados de Muse...' (Most listened to tracks by Muse...):

Tema	Ver más
Muse - Time Is Running Out	00:30
Muse - Hysteria	00:30
Muse - Starlight	00:30
Muse - Knights of Cydonia	00:30
Muse - New Born	00:30
Muse - Plug In Baby	00:30

At the bottom right, there's an advertisement for 'live Player' with the text 'Radio en línea 100% GRATUITA' and a radio icon. The URL 'www.Live-Player.com/Radios' and 'Atenciones Gooooooolle' are also visible.

Figura 3.3: Sección de artistas y recomendaciones en Last.fm

Uno de los mayores atractivos de last.fm es la forma de medir el éxito de una canción, atendiendo al número de personas que reproducen la canción. Incluye además un reproductor descargable por el usuario, y un editor del perfil de usuario.

3.3.2. Pandora (www.pandora.com)



Figura 3.4: Portal de pandora.com

Pandora es una radio online que también se basa en un algoritmo de filtrado colaborativo, recomendando a los usuarios canciones que son del agrado de otros usuarios con gustos similares. Su gran punto fuerte respecto a otras radios es la interfaz, muy intuitiva y fácil de utilizar. El usuario se crea su propia emisora de radio, con la ayuda opcional de un asistente en la que posteriormente puede añadir y editar canciones escogidas por él, además de aquellas canciones que le sean recomendadas.

3.4. Difusión de las redes sociales

Actualmente el uso de las redes sociales en Internet ha aumentado de una manera vertiginosa convirtiéndose en promisorios negocios para empresas, artistas, marcas y sobretodo lugares para encuentros humanos. Por ejemplo, las redes sociales como Facebook o Twitter han tenido tal crecimiento en los últimos años que en la actualidad ocupan los primeros puestos en el número de visitantes diarios en internet. Un dato que nos indicaría claramente los intereses que tienen los usuarios en la red sería el siguiente: según la empresa de medición de tráfico de Internet Experian Hitwise [26], Facebook superó en marzo de 2010 por primera vez en número de visitas a Google en EEUU, el espectacular aumento de Facebook nos indica la creciente evolución de Internet hacia las redes sociales y hacia redes abiertas en las que los usuarios pueden colgar sus contenidos.

Normalmente se utilizan las redes sociales para el uso personal, pero, por ejemplo en Facebook, existen también herramientas para la creación de grupos de usuarios que tienen un interés común, dichos grupos pueden llegar a alcanzar millones de usuarios. Además de este uso personal, las redes sociales se pueden utilizar para promocionar tus actividades, darlas a conocer al mundo entero o simplemente a aquellos que estén interesados en las mismas y es posible realizarlo en forma personal, en forma profesional o inclusive en forma institucional. Un ejemplo de ello es que constantemente podemos informar en Facebook y Twitter sobre la publicación de nuevos artículos y demás temas relacionados de nuestros sitios. Así que las formas de difundir o promocionar nuestras actividades personales, profesionales o institucionales son variadas, ya no solamente se puede usar las redes sociales para comunicarte con los amigos, sino también para encontrar personas con un interés común o interesadas en un tema en particular [25].

Actualmente podemos observar que la mayoría de portales web, por ejemplo portales de periódicos, enlazan con estos sitios siendo así un beneficio para ambos. Por un lado, las redes sociales seguirán teniendo cada vez más auge y podrán llegar a más gente, y por otro lado que mejor forma de dar publicidad a un portal web que haciéndolo desde las propias redes sociales que usan diariamente multitud de

personas. Por tanto, para seguir a la altura de los actuales portales web vamos a enlazar nuestra radio con estas redes sociales, como son Twitter o Facebook, así si el visitante quiere promocionar OL-Radio en su cuenta de Twitter o Facebook lo podrá hacer sin ningún problema con un solo clic. Además, será un objetivo para el presente proyecto construir perfiles de usuarios usando las conexiones que tiene cada usuario con las redes sociales más famosas como Facebook, Twitter, Youtube, Flick-R o LastFM. Para concluir con un ejemplo, un músico podrá subir su música en OL-Radio y promocionarse al mismo tiempo creando un perfil usando sus conexiones con las redes sociales. Así visitando la cuenta de este músico en OL-Radio tendremos enlace directo a sus noticias en las redes sociales (se verá con detalle en el Anexo B: Manual de usuario).

3.5. Ejemplos de redes sociales en Internet

En este apartado presentamos tres redes sociales ampliamente conocidas: Facebook, MySpace y Twitter.

3.5.1. Facebook (www.facebook.com)



Figura 3.5: Ejemplo de un perfil en facebook.com

Facebook es la red social que conecta amigos en todo el mundo ingresando con gran aceptación en el mundo.

En Facebook las personas mantienen contacto con sus amigos, cargan un número ilimitado de fotos, comparten vínculos y vídeos y obtienen información acerca de las personas que van conociendo.

Algunos de la multitud de servicios que ofrece son los siguientes: listas de amigos, crear grupos o páginas, editar tu muro, gestionar tus fotos, disfrutar de sus aplicaciones o jugar a sus juegos.

3.5.2. MySpace (www.myspace.com)

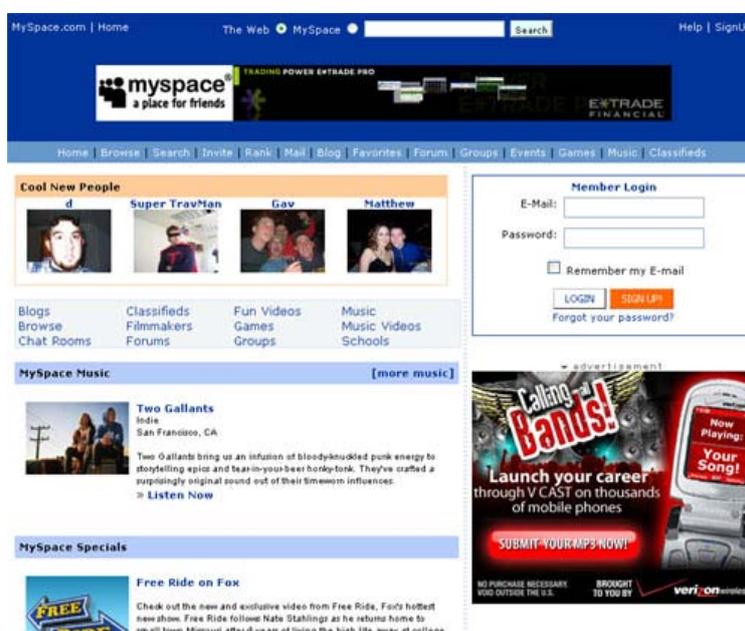


Figura 3.6: Portal de myspace.com

Myspace es un sitio web, de interacción social constituido por perfiles personales de usuarios que incluye redes de amigos, grupos, blogs, fotos, vídeos y música, además de una red interna de mensajería que permite comunicarse a unos usuarios con otros y un buscador interno. Entre sus posibilidades, Myspace ofrece perfiles especiales para músicos y sus usuarios usan el servicio con diversos y diferentes fines, entre ellos el comunicarse con amigos o familiares, el conocer gente, por motivos de trabajo, como ha servido para que grupos musicales se den a conocer, así todos tienen un perfil en la página, siendo a veces más visitada que la verdadera página oficial. Esta red social ha servido de inspiración para una parte del presente proyecto ya que en OL-Radio también se contará con perfiles especiales para músicos y la posibilidad de que puedan compartir sus canciones.

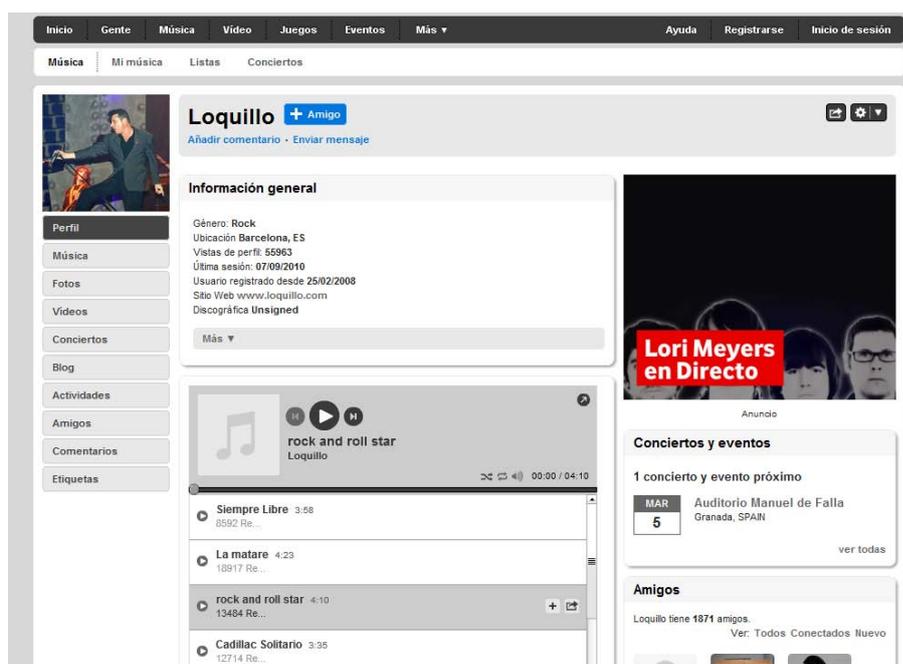


Figura 3.7: Perfil del músico Loquillo en MySpace.com

3.5.3. Twitter (www.twitter.com)

Twitter es un sitio web de “**microblogging**” que permite a sus usuarios enviar y leer microentradas de texto de una longitud máxima de 140 caracteres denominados *tweets*. El envío de estos mensajes se puede realizar tanto por el sitio web de Twitter, como vía SMS (*short message service*) desde un teléfono móvil, desde programas de mensajería instantánea o incluso desde cualquier aplicación de terceros, como por ejemplo Facebook, Xbox 360, Tuenti o Sony Ericsson. Estas actualizaciones se muestran en la página de perfil del usuario, y también se envían de forma inmediata a otros usuarios que han elegido la opción de recibirlas. A estos usuarios se les puede restringir el envío de estos mensajes sólo a miembros de su círculo de amigos o permitir su acceso a todos los usuarios, que es la opción por omisión.

Los usuarios pueden recibir las actualizaciones desde la página de Twitter, vía mensajería instantánea, SMS, RSS y correo electrónico.



Figura 3.7: Perfil del músico Bob Dylan en Twitter.com

3.6. Licencias de libre distribución de contenidos.

Una licencia de Software u otra obra intelectual consiste en la autorización, por parte del autor de dicha obra para utilizarla de una forma establecida y bajo unas restricciones y derechos legales sobre su uso. Un ejemplo de licencia podría ser aquella que restringe el territorio de utilización de un contenido de audio, la duración de dicha utilización o cualquier otra restricción que el autor del contenido estime oportuna.

En general, la persona que crea la obra puede autorizar o restringir el uso, modificación y/o difusión de la misma, adscribiéndose a un tipo de licencia determinado.

Nos centraremos en las licencias de libre distribución, que por lo general permiten la descarga, utilización y difusión de las obras adscritas a ella de forma gratuita y legal. En particular haremos un breve estudio de las siguientes dos licencias:

- *Copyleft*[27]
- *Creative Commons*[18]

3.6.1. Copyleft

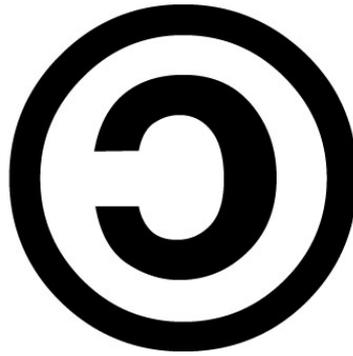


Figura 3.8: Logotipo de la licencia Copyleft

El término *Copyleft* se atribuye a una licencia o grupo de derechos de autor que se caracteriza por la eliminación de las limitaciones de modificación o distribución impuestas por la conocida licencia *Copyright*. La idea es garantizar al receptor de una obra la posibilidad de utilizar, modificar y difundir no sólo la propia obra, sino también posibles versiones derivadas de la misma, modificadas por el propio receptor. La única condición que *Copyleft* impone es que cualquier obra que derive de la modificación de una obra bajo licencia *Copyleft* siga manteniendo dicho régimen de derechos.

Existen varias licencias de tipo *Copyleft*, algunas de ellas son las siguientes:

- **GPL:** La GNU GPL (General Public License) es una licencia creada por la Free Software Foundation a mediados de los 80, y su principal objetivo es proteger la libre distribución, modificación y uso de software, declarando a todo software cubierto por esta licencia como Software Libre.
- **Licencia Arte Libre:** La razón de ser de esta licencia radica en promover y proteger prácticas artísticas liberadas de las reglas de la economía de mercado.
- **Coloriuris:** Esta licencia está destinada a los creadores de contenidos – literarios, musicales, audiovisuales y fotográficos – que utilizan la Web para su difusión, publicación y/o puesta a disposición y que desean ceder los derechos patrimoniales de sus creaciones dentro y fuera de la red.

Hoy en día, existen producciones creativas de prácticamente cualquier tipo (Software, cultura, arte, obras científicas, etc.) sujetas a la licencia *Copyleft*. En la dirección Web <http://fundacioncopyleft.org/> encontramos toda la información relativa a Copyleft.

3.6.2. Creative Commons



Figura 3.9: Logotipo de la licencia Creative Commons

Para realizar este proyecto se ha optado por esta licencia para buscar los archivos musicales que componen nuestra base de datos de canciones. Por esta razón, merece una descripción detallada que pasamos a hacer a continuación.

Creative Commons es una organización no lucrativa, de origen americano, que establece un punto intermedio entre los derechos de propiedad intelectual de *Copyright* y la total eliminación de dichos derechos en licencias como *Copyleft*. Se basa en la idea de que algunos autores pueden no querer ejercer todos los derechos de propiedad intelectual que la ley les permite, ya que se ha comprobado que a veces el total derecho de copia no ayuda a conseguir toda la difusión que ellos desearían.

Muchos usuarios de Internet buscan compartir sus obras y poder reutilizar y modificar obras en común con otros usuarios. *Creative Commons* se ajusta a dicho propósito, ofreciendo a su comunidad un conjunto de licencias sin coste alguno. Estas licencias permiten a los autores otorgar de forma voluntaria a otras personas la facultad de utilizar, copiar, distribuir, exhibir o modificar su obra, sujetos a una serie de condiciones.

Es evidente que Internet ha supuesto el mejor escaparate para extender el uso de las licencias *Creative Commons*, ya sea por medio de blogs, sitios Web multimedia con streaming, wiki, etc.

Podemos encontrar información bastante precisa sobre las licencias *Creative Commons* y el modo de utilizarlas en:

<http://es.creativecommons.org/proyecto/faq.php>

3.6.2.1. Creative Commons en España

Son varios los países que ya han incluido y adaptado el uso de las licencias *Creative Commons* en sus respectivas legislaciones, y muchos otros los que están en proceso. La traducción y adaptación de licencias *Creative Commons* fue presentada al régimen español de Derechos de Autor en octubre de 2004. Hasta esta fecha, la mayor parte de usuarios adaptaban sus obras a un modelo internacional de licenciamiento ajustado a las normas jurídicas norteamericanas, que son en muchos aspectos diferentes a las de España.

Las licencias *Creative Commons* vigentes en España varían según las cuatro condiciones en que el autor quiera compartir su obra:

Reconocimiento (*Attribution*): El material creado por un artista puede ser distribuido, copiado y exhibido por terceras personas si se muestra en los créditos.



No Comercial (*Non Commercial*): El material original y los trabajos derivados pueden ser distribuidos, copiados y exhibidos mientras su uso no sea comercial.



Sin Obra Derivada (*No Derivate Works*): El material creado por un artista puede ser distribuido, copiado y exhibido pero no se puede utilizar para crear un trabajo derivado del original.



Compartir Igual (*Share Alike*): El material creado por un artista puede ser modificado y distribuido pero bajo la misma licencia que el material original.



A partir de las cuatro combinaciones básicas se pueden generar las seis licencias que se pueden escoger. En ningún caso los autores trasladan la autoría de sus producciones, solamente autorizan a otras personas, mediante una serie de privilegios expresos, según las facultades o limitaciones establecidas. Las seis licencias disponibles son las siguientes:



Reconocimiento (by): Se permite cualquier explotación de la obra, incluyendo una finalidad comercial, así como la creación de obras derivadas, la distribución de las cuales también está permitida sin ninguna restricción.



Reconocimiento - NoComercial (by-nc): Se permite la generación de obras derivadas siempre que no se haga un uso comercial. Tampoco se puede utilizar la obra original con finalidades comerciales.



Reconocimiento - NoComercial - CompartirIgual (by-nc-sa):

No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.



Reconocimiento - NoComercial - SinObraDerivada (by-nc-nd):

No se permite un uso comercial de la obra original ni la generación de obras derivadas.



Reconocimiento - CompartirIgual (by-sa):

Se permite el uso comercial de la obra y de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.



Reconocimiento - SinObraDerivada (by-nd):

Se permite el uso comercial de la obra pero no la generación de obras derivadas.

En nuestro proyecto los usuarios de tipo músico (grupos o artistas individuales) que decidan subir sus propias canciones podrán decidir una de las seis licencias de Creative Commons en España anteriormente comentadas. Eso hará que el perfil de los músicos colaboradores con OL-Radio sean músicos que no disponen de discográfica y por tanto necesitan un medio de promoción para sus canciones pero respetando su autoridad sobre sus canciones, al lo que las licencias Creative Commons se adaptan perfectamente.

La música es por excelencia una actividad social, una conversación entre los artistas y el público. Las licencias de Creative Commons le permitirán a los artistas que así lo deseen establecer esta conversación de una forma distinta, de una forma que le permite al músico conservar algunos derechos y entregarle algunos derechos al público oyente.

Existen muchas formas en que un músico puede usar alguna de las licencias Creative Commons. Por ejemplo, un músico puede elegir una licencia *CC-Reconocimiento-NoComercial-SinObraDerivada* si quiere que algunos de sus temas puedan ser descargados y compartidos legalmente, pero sin permitir versiones ni modificaciones de la obra. Otro músico puede entregar algunos de sus temas bajo una licencia *CC-Reconocimiento-NoComercial* que permite que la gente haga sus propias versiones. También hay numerosas discográficas (*netlabels*) que utilizan licencias Creative Commons.

El movimiento Creative Commons está de parte de la música, de parte de los músicos. Creative Commons ayudará a todos los creadores a encontrar una forma creativa de adaptarse a este nuevo escenario en lugar de intentar detenerlo, para que internet pueda continuar trayendo beneficios para todos.

CAPÍTULO 4

Desarrollo del Proyecto

4.1. Descripción del Proyecto

Una vez se han sentado las bases teóricas de los Sistemas de Recomendación, así como de las radios colaborativas y redes sociales existentes Internet, es el momento de entrar con profundidad en el desarrollo de este proyecto.

Este proyecto es la continuación del proyecto llamado **OL-RADIOUJA** que fue realizado por Iván Palomares Carrascosa presentado y defendido el 6 de Julio de 2009, estaba basado en un algoritmo de filtrado colaborativo, para recomendar canciones a los usuarios, siguiendo para todo ello una arquitectura cliente/servidor. Esta radio colaborativa constaba de tres componentes fundamentales, que suponen los tres pilares básicos del desarrollo de ese proyecto:

- Una base de datos, que incluye todos los datos relativos a usuarios del sistema, canciones, puntuaciones de los usuarios sobre las canciones, predicciones...
- Una interfaz para la aplicación, en forma de portal Web, desde la que los usuarios accederán al sistema, escucharán y evaluarán canciones, y recibirán recomendaciones de música a modo de listas de reproducción.
- Un algoritmo de filtrado colaborativo, que se encarga de calcular las predicciones a partir de los datos de puntuaciones, usuarios y canciones existentes en la base de datos.

Como comentamos en el primer capítulo, la anterior radio fue un éxito aun así tenía algunas carencias y deficiencias que podían corregirse. La propuesta del presente proyecto nace de mejorar las anteriores carencias y el propósito de realizar una mayor funcionalidad de la radio, con el fin de poder potenciar el uso de la nueva radio online colaborativa. A continuación detallamos las mejoras y la ampliación de funcionalidades en esos tres pilares anteriores:

- Ampliación de la base de datos por parte de los usuarios, mediante la gestión de un grupo de música o autor donde podrá crear nuevos álbumes, subir

canciones, gestionar sus propias licencias y hacer modificaciones de perfil de usuario músico.

- Crear una nueva interfaz amigable y sobre todo atractiva, intentando que se parezca a las nuevas redes sociales para aumentar su usabilidad. Enlazarla con los sitios web 2.0 de actual auge.

- Mejorar los perfiles de usuario permitiendo establecer conexiones con las principales redes sociales desde el propio perfil de usuario, así un usuario podrá mostrar en su perfil de la radio online sus actividades en las principales redes sociales. Además la posibilidad de enviar y recibir mensajes privados entre usuarios.

- Crear nuevos requerimientos para facilitar el uso de los usuarios, como poder ver las canciones más escuchadas o las canciones más puntuadas.

- Ejecución del algoritmo de filtrado colaborativo automáticamente.

- Crear el nuevo rol de un administrador de la radio el cual se encargará del buen funcionamiento del sitio; será encargado de revisar canciones, podrá borrar canciones o álbumes y ejecutar manualmente el algoritmo de filtrado.

Tras haber expuesto una presentación y una visión general del proyecto, con su propósito y objetivos, una introducción teórica a los Sistemas de Recomendación, radios colaborativas y redes sociales en Internet, es el momento de detallar, en base a técnicas de Ingeniería del Software, el proyecto realizado. La razón de ello es sencilla: estamos ante un proyecto de desarrollo software y, como tal, hemos de seguir la metodología de la Ingeniería del Software.

No existe una definición única y estandarizada para la Ingeniería del Software. Pese a ello, las dos que pasamos a exponer a continuación resultarán perfectamente válidas para nuestro acometido.

La Ingeniería del Software consiste en la construcción de software de calidad con un presupuesto limitado y un plazo de entrega en contextos de cambio continuo.

La Ingeniería del Software es el establecimiento y uso de principios y métodos firmes de ingeniería, para obtener software económico que sea fiable y funcione de manera eficiente en máquinas reales.

La Ingeniería del Software se compone de las siguientes actividades [15]:

- Especificación de requerimientos: Se obtiene el propósito del sistema, así como las propiedades y restricciones impuestas sobre el mismo.
- Análisis del sistema: Se obtiene un modelo del sistema correcto, completo, consistente, claro y verificable.
- Diseño del sistema: Se establecen los objetivos del proyecto y las estrategias a seguir para conseguirlos.
- Implementación: Consiste en la traducción del modelo lógico del sistema a código fuente.
- Pruebas: Verificar y validar el sistema.

En adelante profundizaremos en cómo se han llevado las citadas etapas durante el desarrollo de nuestro proyecto.

4.2. Especificación de Requerimientos

El primer paso en el proceso de Ingeniería del Software debe ser determinar el propósito último del proyecto, las propiedades que debe satisfacer y las restricciones a las que ha sido impuesto. Este es, sin duda, un paso de vital importancia dentro del desarrollo de un proyecto software, ya que sin conocer el propósito del mismo ni establecer las diferentes limitaciones que debe afrontar, será realmente difícil realizar una aplicación software capaz de cumplir dicho propósito.

En un proyecto de ámbito comercial para una empresa real, la forma de determinar el propósito es mediante una serie de estudios como pueden ser: entrevista con los clientes, encuestas con posibles usuarios, observaciones en vivo de la actividad diaria en la empresa, estudios de viabilidad o análisis de la situación actual de la empresa. En nuestro caso, estamos ante un proyecto de ámbito académico, por lo que el propósito es conocido desde la concepción del mismo:

Teniendo una radio colaborativa personalizada, mediante un algoritmo de filtrado colaborativo basado en ítem y provista de una interfaz Web. Dicha radio daba al usuario recomendaciones de música, en forma de lista de reproducción, basada en las preferencias musicales del mismo, y permitiendo en todo momento evaluar canciones, hayan sido o no previamente evaluadas. Ahora la radio deberá dar la posibilidad a los usuarios de subir su propia música a la vez de servirle de un sitio para promocionarse, mejorando los perfiles de usuarios y permitiendo recibir y enviar mensajes; por tanto ahora la radio será más dinámica. Para controlar está subida creamos la figura de un administrador de la radio que revisará las canciones en el momento que sean subidas, así como eliminar en cualquier momento cualquier canción o álbum y ejecutar manualmente el algoritmo de filtrado desde el propio sitio. Por último se mejorará la imagen de la radio con una nueva interfaz, modernizando la imagen y asimilándola con las famosas páginas de redes sociales y posibilitando abundantes conexiones con estas.

Tras haber definido el propósito último del proyecto, el siguiente paso consiste en especificar los requerimientos del mismo. Los requerimientos de un proyecto software son el conjunto de propiedades o restricciones, definidas de forma totalmente

precisa, que dicho proyecto ha de satisfacer. Existen dos tipos bien diferenciados de requerimientos:

- **Requerimientos funcionales:** Aquellos que se refieren expresamente al funcionamiento del sistema.
- **Requerimientos no funcionales:** Son todos aquellos requerimientos no referidos al estricto funcionamiento del sistema, sino a otros factores externos.

En los dos epígrafes siguientes, definiremos estos requerimientos (tanto funcionales como no funcionales), para el proyecto sobre el que versa la presente memoria.

4.2.1. Requerimientos funcionales

Como hemos dicho, los requerimientos funcionales de un sistema software son aquellos que describen las funcionalidades que dicho sistema debe proporcionar a los usuarios del mismo, para cumplir su objetivo.

Al estar ante un proyecto de tipo académico, no disponemos de cliente alguno para obtener los requerimientos, por lo que nos basamos en otros Sistemas de Recomendación de música existentes en el mercado, de reconocido éxito, para establecer los requerimientos.

Como el actual proyecto parte de la mejora y aumento de funcionalidades del proyecto OL-Radio basada en un motor de filtrado colaborativo, en este nuevo proyecto se mantienen las anteriores requerimientos funcionales que se pueden consultar en la memoria de dicho proyecto [28], por tanto, en la presente memoria nos centraremos en los nuevos requerimientos o en un requerimiento funcional que haya sido sufrido algún tipo de cambio. Con esto evitaremos que la presente memoria sea demasiado extensa.

Para una mejor comprensión del lector agruparemos estos requerimientos funcionales según el tipo de usuario:

- **Requerimientos funcionales para usuario de tipo oyente y de tipo músico**
 - Editar el perfil del usuario. Esta funcionalidad tiene las siguientes novedades:
 - Enviar y recibir mensajes privados entre usuarios.
 - Extraer el estado de las conexiones del usuario con las redes sociales.
 - Disponer de información clara acerca de todos los músicos colaboradores, así como de sus álbumes, su biografía y sus últimas noticias en redes sociales.
- **Requerimientos funcionales para usuario de tipo músico**
 - Poder crear álbumes y subir canciones eligiendo el tipo de licencia que desean para gestionarlas y poder editarlos o eliminarlos en cualquier momento.
- **Requerimientos funcionales para usuario de tipo administrador**
 - Revisar las canciones subidas para un correcto funcionamiento de la radio.
 - Ejecutar el algoritmo de filtrado desde el propio sistema manualmente o configurar las ejecuciones automáticas.
 - Eliminar álbumes o canciones que no sean adecuados para nuestro sistema.

4.2.2. Requerimientos No Funcionales

Los requerimientos no funcionales son las restricciones impuestas a los requerimientos funcionales del sistema. Pese a que en principio puedan no parecer demasiado relevantes, son tan importantes como los requerimientos funcionales y, en muchos casos, pueden incluso llegar a ser críticos para la aceptación del sistema. Normalmente, estos requerimientos especifican propiedades del sistema software en sí

y de la interfaz de usuario, además de todas las restricciones impuestas por la organización.

- **Requerimientos del Equipo Informático**

Dado que nuestro proyecto se sustenta sobre una arquitectura cliente/servidor, será necesario distinguir entre los requerimientos del equipo cliente y los requerimientos del servidor.

Los **requerimientos del Equipo Informático del cliente** son bastante sencillos. Tan sólo es necesario un equipo con conexión a Internet (preferiblemente de banda ancha, para optimizar el tiempo de descarga en el cliente de la música a escuchar y el tiempo de subida en caso que el cliente lo desee), que disponga de un navegador adecuado para visualizar la aplicación correctamente. Dicho navegador debe poder permitir la reproducción de contenido Flash (normalmente el navegador avisa e invita al usuario a instalar el plugin en caso de no existir) y, a ser posible, con el bloqueo de pop-ups desactivado. Además, es necesario disponer de altavoces y tarjeta de sonido.

Los **requerimientos del Equipo Informático del servidor**, son bastante más ambiciosos y detallados, por lo que los dividiremos en dos grupos: requerimientos de hardware y requerimientos de software.

- 1) **Hardware del Equipo Servidor**

-**Tipo de Procesador:** Intel Xeon X3320 Quad Core a 2,50 GHz

-**Número de procesadores:** 1

-**Núcleo del procesador disponible:** Quad

-**Memoria caché interna:** 6 MB de caché de nivel 2.

-**Bus del sistema:** Bus frontal de 1333 MHz

-**Memoria de serie:** 4 GB de memoria estandar.

- Unidad de disco duro interna:** 1 unidad SATA 1TB, 3,5" sin conexión en caliente.
- Velocidad de la unidad de disco duro:** 7.200 rpm
- Almacenamiento masivo interno:** 1 TB
- Interfaz de red:** Adaptador de servidor NC105i Gigabit incorporado.
- Puertos de E/S externos:** Serie: 1, USB 2.0: 5, Gráficos:1, Teclado:1,Ratón:1, Conector de red RJ-45.

2) Software del Equipo Servidor

- Sistema Operativo Instalado:** Mandriva v.Spring 2009, Sistema Operativo de libre distribución de la familia Linux.
- Máquina virtual:** Java v1.6.
- PHP:** Versión 5.0.
- Navegador:** Última versión del navegador Mozilla Firefox.
- Sistema Gestor de Bases de datos:** MySql Server 5.1.
- Gestor de Contenidos (CMS):** Drupal v6.10, se trata de la aplicación sobre la que se monta el prototipo de nuestro proyecto.
- Servidor Web:** Apache 2.
- Entorno de desarrollo Java:** NetBeans 6.10 con JDBC.

Parte del software necesario será proporcionado al administrador de la aplicación, el cual dispone de un manual para su instalación en el Anexo A.

4.2.3. Requerimientos de la Interfaz

Los requerimientos de la interfaz gráfica entre la aplicación y el usuario están estrechamente ligados a la usabilidad y sus principios [13]. La usabilidad se puede definir de varias formas:

Usabilidad se define como la medida en que un producto se puede usar por determinados usuarios para conseguir objetivos específicos con efectividad, eficiencia y satisfacción en un contexto de uso específico [19].

Usabilidad se refiere a la capacidad de un software de ser comprendido, aprendido, usado y ser atractivo para el usuario, en condiciones específicas de uso.

La usabilidad puede definirse de forma coloquial como facilidad de uso, ya sea de una página Web, una aplicación informática o cualquier otro sistema que interactúe con el usuario.

Partiendo de estas tres definiciones, se pueden obtener los principios básicos de la usabilidad, los cuales se asociarán a los requerimientos no funcionales que deberá cumplir la interfaz gráfica:

- **Facilidad de aprendizaje:** Se refiere a aquellas características de la interfaz que permiten a los usuarios noveles comprender cómo usarla inicialmente y cómo obtener un máximo grado de productividad. Para que un sistema sea fácil de aprender debe reunir las siguientes características.

- *Predecibilidad:* El conocimiento de la historia de una interacción debe ser suficiente para determinar el resultado de una interacción futura.

- *Síntesis:* El usuario debe poder captar fácilmente los cambios de estado del sistema.

- *Familiaridad:* Es importante diseñar el sistema de forma que los elementos de su interfaz resulten conocidos y familiares para el usuario.

- *Generalización:* Las tareas semejantes deben ser resueltas de modo parecido.

▪ *Consistencia*: Todos los mecanismos que se utilizan deben ser usados siempre de la misma manera, sea cual sea el momento en que se utilicen. Esto se consigue, por ejemplo, siguiendo guías de estilo y diseñando con un *look & feel* común.

• **Flexibilidad**: Este principio de usabilidad establece que debe haber varias formas en que el sistema y el usuario intercambian la información. Las características que hacen a una interfaz flexible son:

▪ *Control del diálogo por parte del usuario*: Hay que proporcionar al usuario la capacidad, siempre que sea posible, para decidir cuándo empezar u acabar las operaciones.

▪ *Migración de tareas*: Tanto el usuario como el sistema deben poder realizar una tarea en exclusiva, o pasarla al otro, o realizarla de forma conjunta.

▪ *Capacidad de sustitución*: La capacidad de sustitución permite que valores equivalente puedan ser sustituidos los unos por los otros. Un ejemplo de ello es el del ancho de página en un procesador de textos, que puede expresarse en centímetros o en pulgadas.

▪ *Capacidad de adaptación*: La interfaz debe poder adaptarse automáticamente a las necesidades del usuario actual.

• **Robustez**: Es el nivel de fiabilidad del sistema, o el grado en que el sistema es capaz de tolerar fallos durante la interacción. Las siguientes características hacen posible la robustez:

▪ *Navegabilidad*: El usuario debe poder observar el estado del sistema sin que esta observación repercuta de forma negativa en él.

▪ *Uso de valores por defecto*: Ayudan al usuario mediante recuerdo pasivo. Mostrar un valor por defecto ayuda a que el usuario sepa qué tipo de valor debe introducir.

▪ *Persistencia*: Un sistema persistente es aquel en el que las notificaciones al usuario permanecen como objetos manipulables después de su presentación.

- *Recuperación de información:* La aplicación debe permitir el regreso a un estado anterior, tras algún error en una interacción previa.

- *Tiempo de respuesta:* Es el tiempo que necesita el sistema para reflejar los cambios realizados por el usuario.

4.3. Análisis del Sistema

Tras haber conocido el propósito del proyecto software, las propiedades de este y las restricciones a las que debe someterse, es el momento de analizar el sistema y crear un modelo del mismo que sea correcto, completo, consistente, claro y verificable. Para ello, se definirán los casos de uso según los requerimientos previamente obtenidos y, acto seguido, se describirán los principales escenarios y flujos de eventos de dichos casos de uso [15], se van a analizar todos los casos de uso y escenarios que tiene el proyecto.

4.3.1. Modelo de Casos de Uso

Un caso de uso representa una funcionalidad concreta dada por el sistema como un flujo de eventos. También se puede definir como la representación de una situación o proceso de interacción de un usuario con la aplicación o sistema.

Los casos de uso son tareas con significado, coherentes y con cierta independencia, que los actores realizan de forma cotidiana al utilizar el sistema. En un caso de uso pueden participar uno o varios actores.

En definitiva, los casos de uso explican cómo se realiza una tarea de forma precisa, y constan de los siguientes elementos:

- Nombre único e unívoco del caso de uso.
- Actores participantes.
- Condiciones de entrada.

- Flujo de eventos.
- Condiciones de salida.
- Requerimientos especiales.

Por lo tanto, la primera cuestión es la de determinar cuáles son los actores participantes en cada uno de los casos de uso.

Un actor modela una entidad externa que interacciona con el sistema, es decir, es un tipo de usuario del sistema. Al igual que ocurre con un caso de uso, un actor debe tener un nombre único y adecuado, que puede ir acompañado de una descripción del mismo.

En nuestro sistema contamos con los actores siguientes:

Usuario_registrado: Se corresponde con cada uno de los usuarios que se han registrado en el sistema. Este usuario se dividirá en dos tipos de usuarios.

- **Oyente:** Este usuario podrá realizar tareas propias como escuchar una radio personalizada, evaluar canciones, modificar sus datos personales y de perfil, etc.
- **Músico:** Podrá realizar exactamente las mismas tareas que un usuario oyente con la diferencia de que éste podrá crear álbumes, subir sus propias canciones, gestionar sus licencias, editar los álbumes y eliminar álbumes o canciones propias.

Usuario_anónimo: Se refiere a cada uno de los usuarios, no registradas en el sistema, que pueden utilizar los servicios básicos proporcionados por este, tales como escuchar una radio global, visualizar los álbumes existentes, músicos existentes, estadísticas, etc.

Administrador_radio: Este usuario solamente podrá tener acceso a la parte de gestión del sistema es decir, revisión de las canciones subidas, ejecuciones al algoritmo de filtrado y eliminaciones de álbumes o canciones.

BBDD (Base de Datos): Es la base de datos que almacena y proporciona los datos necesarios para el sistema (usuarios registrados y datos de su perfil, canciones, puntuaciones de usuarios sobre canciones, predicciones para los usuarios sobre canciones, listas de reproducción personalizadas y datos del sitio Web).

Una vez definidos cuáles van a ser los actores del sistema, llega el momento de crear los diferentes casos de uso. Para ello, es importante que cada uno de los requerimientos funcionales anteriormente definidos sean cubiertos por al menos uno de los casos de uso aunque, por otro lado, puede haber casos de uso nuevos, en los que no aparezca ninguno de los requerimientos, ya que nos encontramos en una fase de refinamiento del sistema en la que pretendemos construir un modelo detallado del mismo.

El primer paso para la realización del modelo de casos de uso de nuestro sistema es la obtención de los diversos diagramas de casos de uso de nuestro sistema. El primero de ellos es un diagrama frontera, es decir, un diagrama que describe completamente la funcionalidad de un sistema.

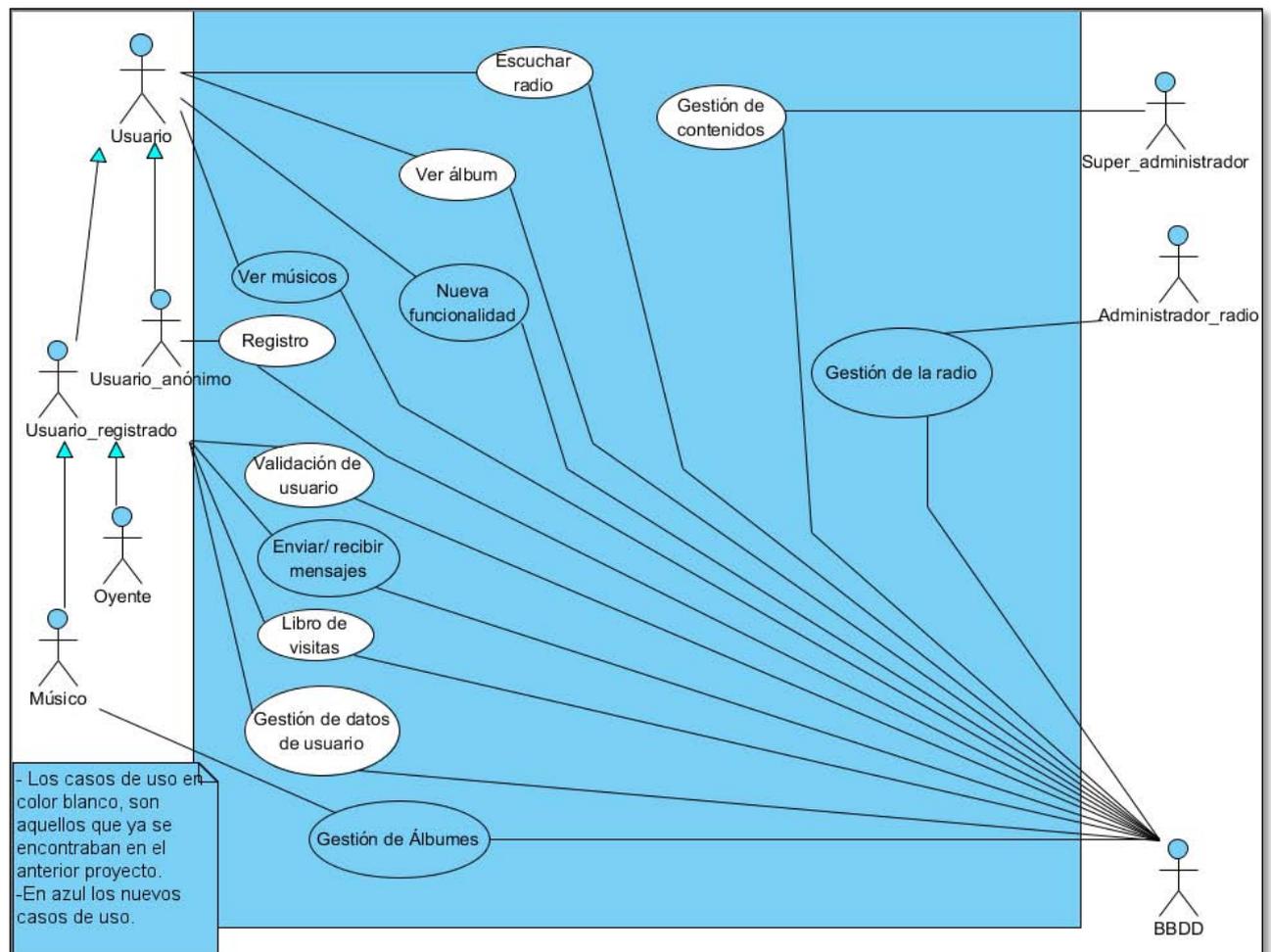


Figura 4.1: Diagrama Frontera del sistema

Los casos de uso reflejados en un diagrama frontera pueden ser lo suficientemente precisos o, por el contrario, necesitar ser explicados en mayor detalle. A la hora de detallar un caso de uso se pueden emplear dos tipos de relaciones.

- **<<extend>>**: Se trata de una relación cuyo sentido es hacia el caso de uso a detallar, que representa comportamientos excepcionales del caso de uso.
- **<<include>>**: Es una relación cuyo sentido es contrario al de la relación <<extend>>, que representa un comportamiento común del caso de uso.

En el caso de nuestro sistema nos encontramos ante varios casos de uso que deben ser descritos con mayor profundidad.

A continuación, describimos detalladamente los nuevos casos de uso mostrados en azul en la figura 4.1 o aquellos casos de uso que ya se encontraban en el anterior proyecto pero han sufrido algún cambio con respecto al anterior proyecto, además exponemos los diagramas que se identifican con aquellos casos de uso que necesitan un mayor nivel de detalle en su descripción.

▪ **Caso de Uso 1: Registro**

Actores participantes: Usuario_anónimo, BBDD.

Condiciones de entrada: Existe usuario que pretende darse de alta en el sistema.

Flujo de eventos:

1. El usuario inicia la aplicación.
2. El sistema muestra un enlace para el registro de usuarios.
3. El usuario accede a la página de registro.
4. El sistema muestra un formulario de entrada.
5. El usuario introduce un identificador de usuario, dirección de correo-e y elige el tipo de usuario que desea ser.
6. El sistema comprueba que los datos introducidos son válidos (E-1) (E-2).
7. Los datos del nuevo usuario quedan almacenados en la BBDD.
8. El usuario recibe un e-mail con la contraseña para validarse en el sistema.

Condiciones de salida: El nuevo usuario ha sido registrado en la BBDD del sistema.

Excepciones

E-1: El identificador de usuario ya está asignado. El sistema informa al usuario de dicha situación. El usuario puede introducir otro identificador de usuario diferente o salir del caso de uso.

E-2: La dirección de correo-e no es válida. El sistema informa al usuario de dicha situación. El usuario puede introducir otra dirección de correo-e diferente o salir del caso de uso.

▪ **Caso de Uso 2:** Gestión de Datos del Usuario

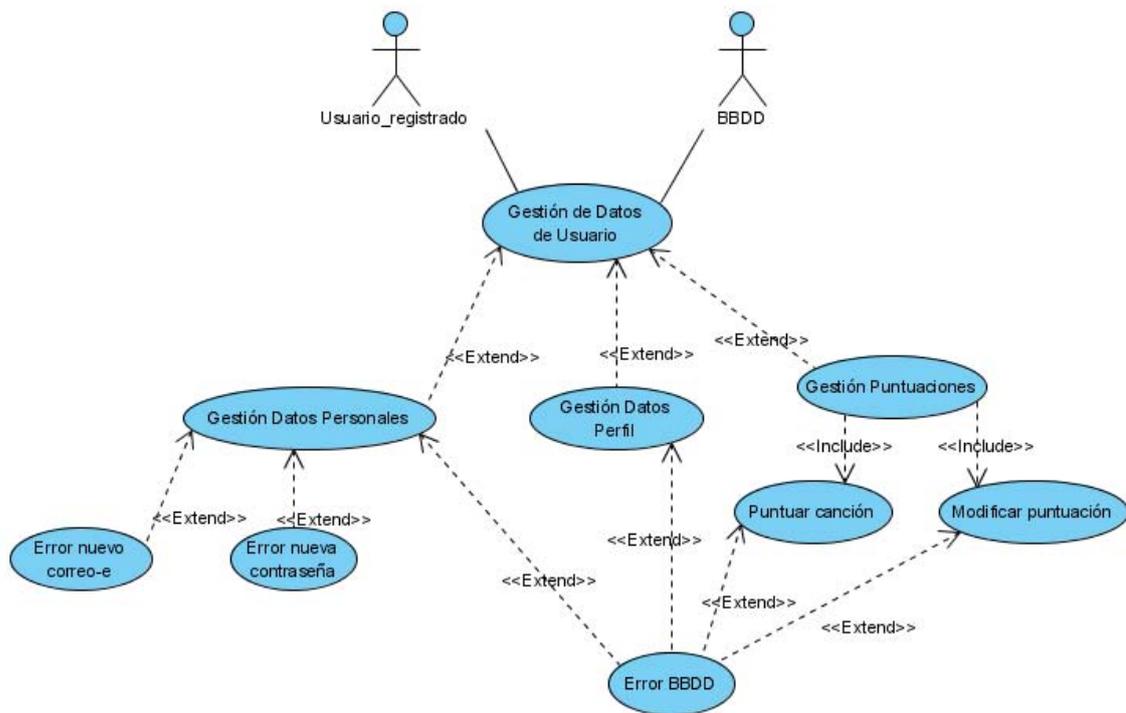


Figura 4.2: Caso de uso 'Gestión de Datos de Usuario'

Actores participantes: Usuario registrado, BBDD.

Condiciones de entrada: El usuario ha entrado en el sistema y se ha validado como usuario registrado.

Flujo de eventos:

1. El usuario desea gestionar sus datos personales, se realiza S-1.
2. El usuario desea gestionar los datos de su perfil, se realiza S-2
3. El usuario desea gestionar sus puntuaciones, se realiza S-3.

Subflujos de eventos:

-S1: Gestión Datos Personales

- 1.1. El usuario elige la opción Mi Cuenta del menú principal.
- 1.2. El sistema se comunica con la base de datos (E-1) para obtener los datos personales del usuario.
- 1.3. El sistema muestra los datos personales del usuario.

1.4. El usuario, si lo desea, modifica su dirección de correo electrónico (E-2), su contraseña (E-3), su género musical favorito o puede añadir alguna conexión con las redes sociales. Si es usuario de tipo músico puede modificar su biografía, estilo de música y sitio web.

1.5. El sistema guarda los cambios efectuados.

-S2: Gestión Datos Perfil

2.1. El usuario elige la opción 'Mis Puntuaciones' del menú principal.

2.2. El sistema se comunica con la base de datos (E-1) para obtener los datos del perfil del usuario.

2.3. El sistema muestra los datos de las puntuaciones hechas anteriormente.

2.4. El usuario, si lo desea, modifica los datos de su perfil (género musical).

2.5. El sistema guarda los cambios efectuados.

-S-3: Gestión Puntuaciones

3.1. El usuario accede a una canción (radio o álbumes).

3.2. El sistema muestra la canción al usuario.

3.3. El usuario elige puntuar la canción.

3.4. El sistema muestra un panel de puntuación asociado a la canción.

3.5. El usuario asigna una puntuación numérica a la canción.

3.6. El sistema se comunica con la base de datos (E-1) para guardar la puntuación.

3.7. El usuario elige la opción Mis Puntuaciones.

3.8. El sistema se comunica con la base de datos (E-1) para obtener los datos de las puntuaciones hechas por el usuario.

3.9. El usuario modifica una de las puntuaciones anteriormente realizadas.

3.10. El sistema se comunica con la base de datos (E-1) para guardar los cambios.

Condiciones de salida: El usuario ha realizado las gestiones oportunas sobre sus datos.

Excepciones:

E-1: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el punto en que se quedó.

E-2: Error en el correo electrónico nuevo. El sistema informa al usuario de dicha situación. El usuario puede introducir una nueva dirección o bien terminar el caso de uso.

E-3: Error en la nueva contraseña. El sistema informa al usuario de dicha situación. El usuario puede introducir una nueva contraseña o bien terminar el caso de uso.

▪ **Caso de Uso 3: Ver nueva funcionalidad**

Actores participantes: Usuario, BBDD.

Condiciones de entrada: El usuario ha accedido al sistema.

Flujo de eventos:

1. El usuario selecciona la opción 'Estadísticas', se realiza S-1.
2. El usuario selecciona la opción 'Quienes somos', se realiza S-2.

Subflujos de eventos:

-S1: Ver estadísticas

- 1.1. El sistema se comunica con la base de datos (E-1) y muestra por defecto el criterio las cinco canciones más escuchadas.
- 1.2. El usuario visualiza la lista y puede acceder si lo desea al álbum de las cinco canciones que quiera.
- 1.3. El usuario puede seleccionar otro criterio de estadísticas.
- 1.4. El sistema se comunica con la base de datos (E-1) para obtener los datos del nuevo criterio.
- 1.5. El sistema muestra la información y las cinco canciones con el nuevo criterio.

-S2: Ver Quienes somos

- 2.1. El usuario selecciona la opción 'Quienes somos'.
- 2.2. El sistema muestra la información sobre las personas implicadas en la realización de este proyecto.
- 2.3. El usuario visualiza esta información.

Condiciones de salida: El usuario ha sido capaz de visualizar la nueva funcionalidad sin problemas.

Excepciones:

E-1: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 1.1.

▪ Caso de Uso 4: Ver músicos

Actores participantes: Usuario, BBDD.

Condiciones de entrada: El Usuario ha accedido al sistema.

Flujo de eventos:

1. El usuario selecciona la opción 'Músicos'.
2. El sistema se comunica con la base de datos (E-1) y muestra una relación de los músicos colaboradores en el sistema.
3. El usuario desea conocer información del músico, se realiza S-1.
4. El usuario desea ver música y noticias de un músico, se realiza S-2.

Subflujos de eventos:

-S1: Visualizar la información del músico

- 1.1. El usuario elige la opción Info.
- 1.2. El sistema se comunica con la base de datos (E-1) para obtener los datos personales del músico.
- 1.3. El sistema muestra el perfil del músico, con su foto, conexiones con redes sociales y otros datos de información general si los tiene, como biografía, estilo musical y sitio web.
- 1.4. El usuario visualiza la información y puede elegir cualquier enlace de las conexiones de este con las redes sociales.
- 1.5. El usuario decide mandar un mensaje privado a este músico, ver caso de uso 5: Enviar/recibir mensaje.

-S2: Visualizar los álbumes del músico

- 2.1. El usuario elige la opción 'Música/Noticias'.
- 2.2. El sistema se comunica con la base de datos (E-1) para obtener los álbumes del músico.
- 2.3. El sistema muestra una foto del músico y la lista de los álbumes que ha creado este músico.
- 2.4. El usuario visualiza la información.
- 2.5. El usuario quiere ver las últimas noticias de este usuario músico y elige la opción 'Últimas Noticias'.

2.6. El sistema se comunica con la base de datos (E-1) y muestra las últimas noticias del músico seleccionado.

Condiciones de salida: El usuario ha visualizado la información del músico deseado correctamente.

Excepciones:

E-1: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 2.

▪ Caso de Uso 5: Enviar/recibir mensajes

Actores participantes: Usuario_registrado, BBDD.

Condiciones de entrada: El Usuario se ha registrado y validado en el sistema.

Flujo de eventos:

1. El usuario desea enviar un mensaje privado, se realiza S-1.
2. El usuario quiere leer los mensajes recibidos, se realiza S-2.

Subflujos de eventos:

-S1: Enviar un mensaje privado

- 1.1. El usuario elige la opción 'Enviar mensaje privado' o 'Escribir mensaje nuevo'.
- 1.2. El sistema muestra un formulario para el envío de un mensaje privado.
- 1.3. El usuario rellena el nombre del usuario o usuarios a los que quiere enviar, el cuerpo del mensaje y un asunto si lo desea.
- 1.4. El usuario puede visualizar el mensaje antes de enviarlo y enviarlo o enviarlo directamente.
- 1.5. El sistema introduce en la base de datos el mensaje privado (E-1) en la lista de mensajes del usuario o usuarios al que se le ha enviado.

-S2: Leer mensajes recibidos

- 2.1. El usuario accede a la bandeja de entrada de sus mensajes privados.
- 2.2. El sistema se comunica con la base de datos (E-2) para obtener la lista de todos los mensajes privados.
- 2.3. El usuario accede al mensaje que desea leer.
- 2.4. El sistema se comunica con la base de datos (E-3) para obtener el contenido del mensaje privado seleccionado.
- 2.5. El usuario recibe la información para leer el mensaje privado.

Condiciones de salida: El usuario ha enviado mensajes privados a otros usuarios o ha podido leer los recibidos sin problemas.

Excepciones:

E-1: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 1.2.

E-2: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 2.1.

E-3: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 2.2.

▪ **Caso de Uso 6: Gestión de álbumes**

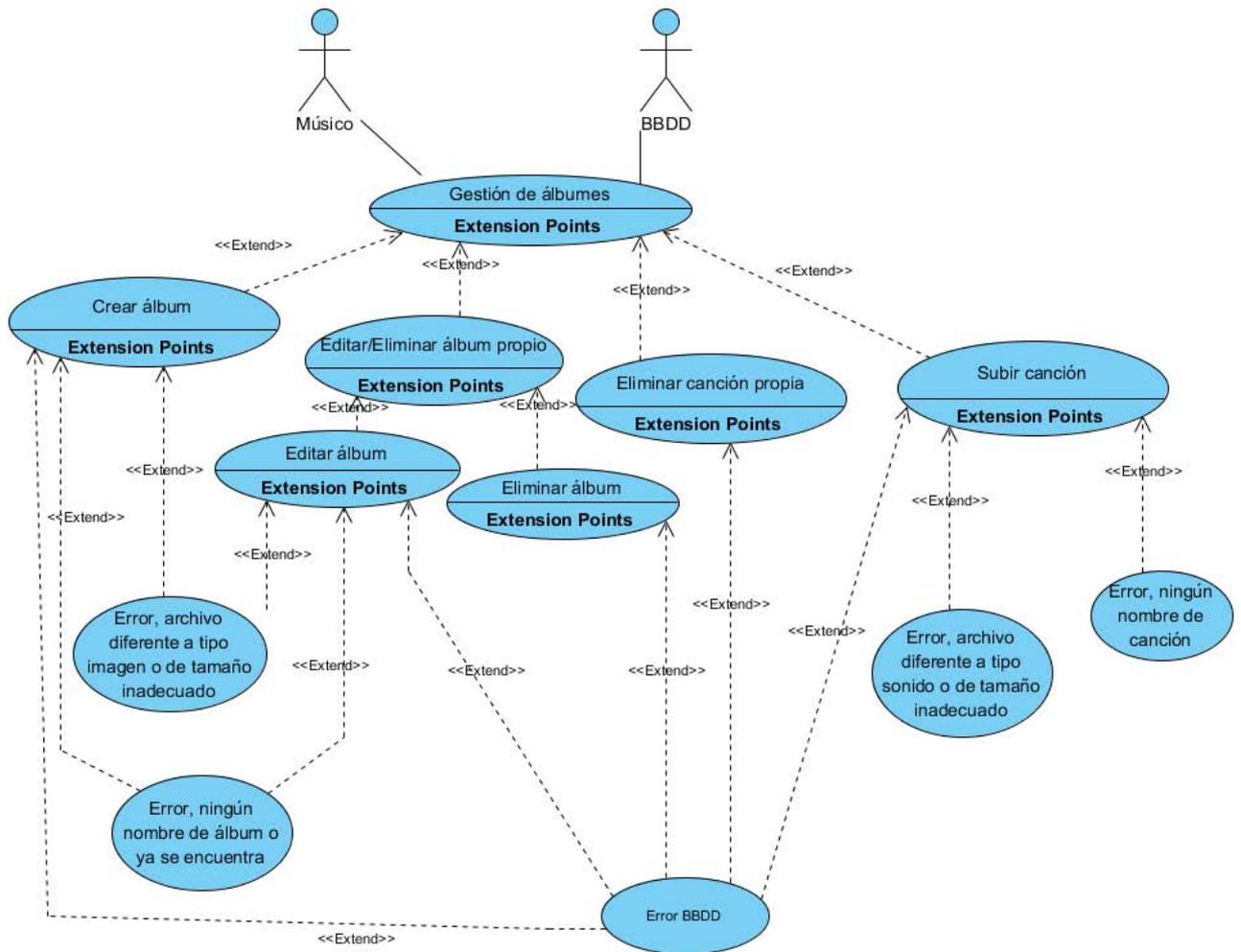


Figura 4.3: Caso de uso 'Gestión de Álbumes'.

Actores participantes: Música, BBDD.

Condiciones de entrada: El Usuario se ha registrado como músico y validado en el sistema.

Flujo de eventos:

1. El usuario elige la opción 'Subir álbum' en el menú de Música, se realiza S-1.
2. El usuario elige la opción 'Mis álbumes' del menú Música para subir canción, se realiza S-2.
3. El usuario elige la opción 'Mis álbumes' del Menú para eliminar canción propia, se realiza S-3.
4. El usuario elige la opción 'Mis álbumes' del menú Música para eliminar un álbum propio, se realiza S-4.
5. El usuario elige la opción 'Mis álbumes' del menú Música para editar un álbum propio, se realiza S-5.

Subflujos de eventos:

-S1: Crear álbum

- 1.1. El sistema muestra un formulario para que el usuario pueda crear su álbum.
- 1.2. El usuario debe escribir un título para su álbum (E-1), elegir el género musical que más se adapte a su álbum, el tipo de licencia oportuna y si lo desea una imagen que servirá como portada de su álbum (E-2).
- 1.3. El usuario selecciona la opción para guardar el nuevo álbum.
- 1.4. El sistema se comunica con la base de datos (E-3) y guarda los datos del nuevo álbum.

Condiciones de salida: El usuario ha creado un nuevo álbum sin problemas.

-S2: Subir canción

Condiciones de entrada: El usuario tiene creado al menos un álbum.

- 2.1. El sistema se comunica con la base de datos (E-4), para obtener los álbumes creados por el usuario.
- 2.2. El usuario selecciona un álbum de los que ha creado.
- 2.3. El sistema se comunica con la base de datos (E-5), para obtener la información necesario al álbum seleccionado.
- 2.4. El usuario introduce un nombre para la canción (E-6) y sube el archivo de tipo sonido (E-7).
- 2.5. El sistema se comunica con la base de datos (E-8), y guarda la nueva canción en la base de datos.

Condiciones de salida: El usuario ha subido una nueva canción a su álbum sin problemas.

-S3: Eliminar canción propia

Condiciones de entrada: El usuario tiene creado al menos un álbum y ha subido al menos una canción.

- 3.1. El sistema se comunica con la base de datos (E-9), para obtener los álbumes creados por el usuario.
- 3.2. El usuario selecciona un álbum de los que ha creado.
- 3.3. El sistema se comunica con la base de datos (E-10), para obtener la información necesaria al álbum seleccionado.
- 3.4. El usuario elige la opción eliminar en la canción que desea eliminar.
- 3.5. El sistema advierte al usuario de que si realiza esta operación eliminará la canción y esta acción no podrá deshacerse.
- 3.6. El usuario decide eliminarla de todos modos.
- 3.7. El sistema se comunica con la base de datos (E-11), para eliminar la canción seleccionada.
- 3.8. El sistema vuelve al paso número 3.3.

Condiciones de salida: El usuario ha eliminado las canciones que anteriormente subió sin problemas.

-S4: Eliminar álbum

Condiciones de entrada: El usuario tiene creado al menos un álbum.

- 4.1. El sistema se comunica con la base de datos (E-12), para obtener los álbumes creados por el usuario.
- 4.2. El usuario selecciona un álbum de los que ha creado.
- 4.3. El sistema se comunica con la base de datos (E-13), para obtener la información necesario al álbum seleccionado.
- 4.4. El usuario elige la opción 'Eliminar Álbum'.
- 4.5. El sistema advierte al usuario de que si realiza esta operación eliminará el álbum con todo su contenido y esta acción no podrá deshacerse.
- 4.6. El usuario decide eliminar el álbum.
- 4.7. El sistema se comunica con la base de datos (E-12) para eliminar el álbum y su contenido.
- 4.8. El sistema vuelve al paso 4.1.

Condiciones de salida: El usuario ha eliminado el álbum seleccionado sin problemas.

-S5: Editar álbum

Condiciones de entrada: El usuario tiene creado al menos un álbum.

- 5.1. El sistema se comunica con la base de datos (E-14), para obtener los álbumes creados por el usuario.
- 5.2. El usuario selecciona un álbum de los que ha creado.
- 5.3. El sistema se comunica con la base de datos (E-15), para obtener la información necesario al álbum seleccionado.
- 5.3. El usuario elige la opción 'Editar Álbum'.
- 5.4. El sistema muestra un formulario para cambiar las opciones del álbum seleccionado.
- 5.5. El usuario puede cambiar el título del álbum (E-16), cambiar el género musical, el tipo de licencia y la una imagen que servirá como portada de su álbum (E-17).
- 5.6. El sistema se comunica con la base de datos (E-18) para editar el álbum seleccionado.
- 5.7. El sistema vuelve al paso número 5.3.

Condiciones de salida: El usuario ha editado el álbum seleccionado sin problemas.

Excepciones:

E-1: El usuario no introduce ningún título para su álbum o el que ha introducido ya se encuentra registrado por otro usuario. El caso de uso se reanuda desde el paso 1.2.

E-2: El usuario sube un archivo que no es de tipo imagen o la imagen que sube no es del tamaño deseado. El álbum se creará igualmente pero sin una imagen, para su visualización se utilizará una imagen por defecto.

E-3: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 1.2.

E-4: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 2.

E-5: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 2.2.

E-6: El usuario no introduce ningún nombre para la canción. El caso de uso se reanuda desde el paso 2.3.

E-7: El usuario sube un archivo que no es de tipo sonido o el que ha subido no es del tamaño adecuado. El caso de uso se reanuda desde el paso 2.3.

E-8: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 2.3.

E-9: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 3.

E-10: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 3.2.

E-11: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 3.5.

E-12: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 4.

E-13: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 4.1.

E-14: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 5.

E-15: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 5.1.

E-16: El usuario no introduce ningún título o el nuevo título ya se encuentra registrado por otro usuario. El caso de uso se reanuda desde el paso 5.4.

E-17: El usuario sube un archivo que no es de tipo imagen o no tiene el tamaño deseado. El caso de uso se reanuda desde el paso 5.4.

E-18: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 5.4.

▪ **Caso de Uso 7: Gestión de la radio**

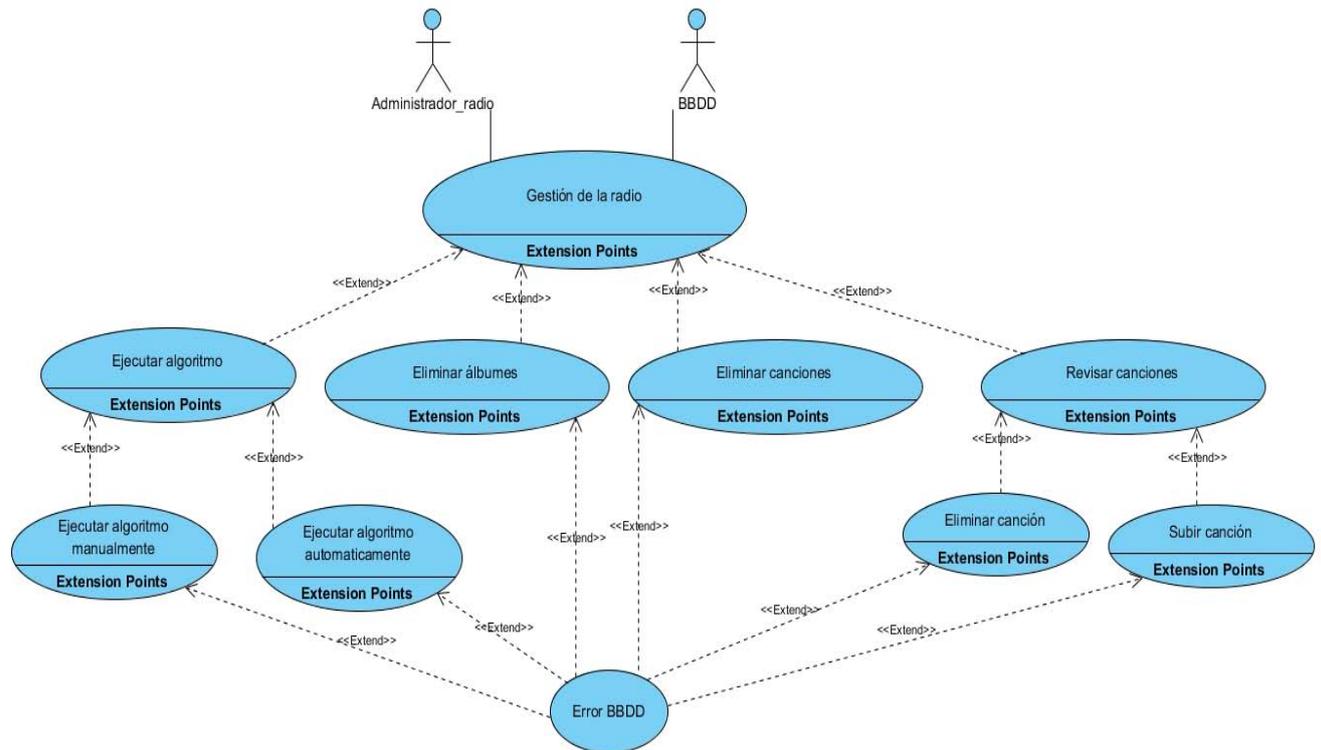


Figura 4.4: Caso de uso 'Gestión de la radio'.

Actores participantes: Administrador_radio, BBDD.

Condiciones de entrada: El usuario esta registrado como administrador_radio y validado en el sistema.

Flujo de eventos:

1. El sistema se comunica con la base de datos (E-1), para obtener la lista de las canciones subidas que aún no han sido revisadas.
2. El usuario decide eliminar una de las canciones pendientes de revisar, se realiza S-1.
3. El usuario decide subir alguna de las canciones pendientes de revisar, se realiza S-2.
4. El usuario elige la opción 'Ejecutar algoritmo' en el menú Administrar radio, se realiza S-3.
5. El usuario elige la opción 'Eliminar álbum' en el menú Administrar radio, se realiza S-6.
6. El usuario elige la opción 'Eliminar canciones' en el menú Administrar radio, se realiza S-7.

Subflujos de eventos:

-S1: Eliminar canción

- 1.1. El usuario selecciona la canción o canciones que desea eliminar y elige la opción 'Eliminar'.
- 1.2. El sistema se comunica con la base de datos (E-1) para eliminar la canción o canciones seleccionadas.

1.3. El sistema vuelve al paso 1.

Condiciones de salida: El usuario ha eliminado algunas de las últimas canciones subidas.

-S2: Subir canción

2.1. El usuario selecciona la canción o canciones que desea subir y elige la opción 'Subir'.

2.2. El sistema se comunica con la base de datos (E-1) para añadir las a la lista de canciones seleccionadas.

2.3. El sistema vuelve al caso de uso 1.

Condiciones de salida: El usuario ha subido algunas de las últimas canciones subidas.

-S3: Ejecutar algoritmo

3.1. El sistema se comunica con la base de datos (E-2), muestra un formulario para configurar la hora y el día para que se ejecute el algoritmo de forma automática e información para ejecutarlo de forma manual. Además obtiene los detalles de la última ejecución del algoritmo recuperando la fecha, hora y el usuario que lo ejecutó.

3.2. El usuario decide ejecutar el algoritmo de forma manual, se realiza S-4.

3.3. El usuario decide cambiar las opciones de la ejecución automática, se realiza S-5.

-S4: Ejecutar algoritmo manualmente

4.1. El usuario elige la opción 'Ejecutar Algoritmo'.

4.2. El sistema se comunica con la base de datos (E-2), para actualizar los datos de predicciones y listas de reproducción que utiliza el algoritmo además de guardar la fecha, hora y el nombre del usuario.

4.3. El sistema muestra un mensaje para que el usuario conozca que la ejecución ha ido correctamente.

Condiciones de salida: El usuario ha ejecutado manualmente el algoritmo sin problemas.

-S5: Ejecutar algoritmo automáticamente

5.1. El usuario decide configurar la ejecución automática del algoritmo y introduce una nueva hora y selecciona un día de la semana o diariamente. Acto seguido, elige la opción 'Cambiar'.

5.2. El sistema se comunica con la base de datos (E-2), actualizando así la nueva configuración para las ejecuciones automáticas del algoritmo.

5.3. El sistema vuelve al punto 3.1 mostrando la nueva configuración para la ejecución automática que el usuario acaba de elegir.

Condiciones de salida: El usuario ha cambiado las opciones de ejecuciones del algoritmo de filtrado de forma automática.

-S6: Eliminar álbum

- 6.1. El sistema se comunica con la base de datos (E-3), para obtener la lista de los álbumes.
- 6.2. El usuario selecciona el álbum que desea eliminar.
- 6.3. El sistema advierte al usuario de que si realiza esta operación eliminará el álbum con todo su contenido y esta acción no podrá deshacerse.
- 6.4. El usuario decide eliminar el álbum.
- 6.5. El sistema se comunica con la base de datos (E-4) para eliminar el álbum y su contenido.
- 6.6. El sistema vuelve al caso de uso 6.1.

Condiciones de salida: El usuario ha eliminado un álbum sin problemas.

-S7: Eliminar canciones

- 7.1. El sistema se comunica con la base de datos (E-5), para obtener la lista de los álbumes.
- 7.2. El usuario selecciona el álbum para ver las canciones de su interior.
- 7.3. El sistema se comunica con la base de datos (E-6), para obtener la lista de canciones pertenecientes al álbum seleccionado.
- 7.4. El usuario selecciona la canción o canciones que desea eliminar y elige la opción Eliminar.
- 7.5. El sistema se comunica con la base de datos (E-6) para eliminar la canción seleccionada.
- 7.6. El sistema vuelve al caso de uso 7.1.

Condiciones de salida: El usuario ha canciones sin problemas.

Excepciones:

E-1: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 1.

E-2: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 3.1.

E-3: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 6.

E-4: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 6.1.

E-5: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 7.

E-6: Ha habido un error en la comunicación entre el sistema y la base de datos. El sistema informa al usuario de dicha situación. El caso de uso se reanuda desde el paso 7.1.

4.3.2. Escenarios

Un caso de uso es una representación abstracta, una abstracción, de una funcionalidad del sistema a realizar. La representación concreta de un caso de uso se realiza mediante la creación de uno o más escenarios que muestren todas las posibles interacciones entre el sistema y los usuarios.

Los escenarios son historias ficticias que describen posibles interacciones de un caso de uso. Son de gran utilidad porque permiten a los diseñadores anticiparse a los problemas. A pesar de ser historias ficticias, deben desarrollarse con el mayor nivel de detalle posible. Deben ser situados en un contexto real con las restricciones que ello supone, etc. De esta manera es mucho más sencillo para los diseñadores discutir sobre la interfaz ya que a las personas les cuesta más trabajo discutir las posteriores decisiones de diseño sobre una situación abstracta.

Un escenario está formado por los siguientes elementos:

- Un nombre único y unívoco
- Una descripción
- Los actores participantes.
- El flujo de eventos.

Como se ha indicado, para cada caso de uso puede haber varios escenarios. Para nuestro proyecto se han creado y descrito gran cantidad de casos de uso, por lo que nosotros vamos a definir solo algunos de los escenarios para los principales casos de uso, así se tendrá un ejemplo de las principales funcionalidades del sistema: registro de usuario, ver músicos, crear álbum, subir canción, editar álbum propio, eliminar álbum propio, revisar canciones, ejecutar algoritmo, eliminar álbum, eliminar canciones y ver estadísticas. Como hemos dicho con anterioridad, aquí nos centraremos en los casos de uso de nueva funcionalidad o aquellos que hayan sufrido algún cambio, para conocer la totalidad de los casos de uso y escenarios es necesario remitirse a la memoria del anterior proyecto [28].

Escenario 1: Registro de usuario

Nombre: Registrar usuario.

Descripción: El usuario Paco entra en el sistema y decide darse de alta en el mismo para poder subir las canciones de su grupo.

Actores participantes: Usuario anónimo y Base de Datos.

Flujo de eventos:

1. Paco entra como invitado en el sistema y pulsa sobre la opción 'Regístrate'.
2. El sistema muestra un formulario con todos los datos requeridos para el registro.
3. El usuario rellena los siguientes campos:
 - a. Usuario: Paco.
 - b. E-mail: Paco@hotmail.es.
 - c. Tipo de usuario: músico.
4. Paco pulsa sobre el botón 'Registrarse' en la parte inferior del formulario.
5. El sistema almacena los datos de Paco en la BBDD, muestra un mensaje informando de que el registro ha sido llevado a cabo con éxito, y envía a la dirección de correo de Paco un mensaje que contiene la contraseña y el enlace que le permitirá activar la cuenta.
6. Paco accede a su buzón de correo, abre el correo enviado por el sistema y pulsa sobre el enlace que aparece en el e-mail.
7. El sistema activa la cuenta de Paco, para que este pueda acceder al sistema como usuario registrado.

Escenario 2: Ver músicos

Nombre: Ver información concreta del músico 'Petete'.

Descripción: El usuario registrado en el sistema con el nombre Antonio quiere ver información sobre el músico 'Petete' para saber los álbumes ha creado en el sistema.

Actores participantes: Antonio y Base de Datos.

Flujo de eventos:

1. El usuario entra en el sistema y se valida de forma correcta como Antonio.
2. El sistema se comunica con la base de datos para obtener las puntuaciones y las predicciones hechas para Antonio y obtiene una lista de reproducción para él.
3. Antonio ignora por el momento su radio y escoge la opción 'Músicos' del menú Radio.
4. El sistema se comunica con la base de datos para obtener y mostrar a Antonio una lista de todos los músicos que colaboran con la aplicación.
5. Antonio elige la opción 'Música/Noticias'.
6. El sistema se comunica de nuevo con la base de datos para obtener los álbumes del músico seleccionado, el sistema muestra una foto, las opciones de 'Enviar mensaje privado', 'Últimas noticias' y debajo la lista de los álbumes que ha creado el músico.
7. El usuario selecciona el álbum 'Haciendo ruido' del músico 'Petete'.
8. El sistema muestra información del álbum concreto 'Haciendo ruido'.

Escenario 3: Crear álbum

Nombre: Crear el álbum 'Mis canciones'.

Descripción: El usuario registrado en el sistema como músico y con el nombre Paco desea crear el álbum 'Mis canciones' para compartir su música con el resto de usuarios del sistema.

Actores participantes: Paco y Base de Datos.

Flujo de eventos:

1. El usuario entra en el sistema y se valida de forma correcta como Paco.
2. El sistema se comunica con la base de datos y al comprobar que se trata de un usuario de tipo músico obtiene la lista de los álbumes creados por él.
3. Paco escoge la opción 'Subir álbum' del menú Mi música.
4. El sistema mostrará a Paco un formulario con los datos requeridos para la creación de un nuevo álbum.
5. Paco rellena los siguientes campos:
 - a. Título: Mis canciones.
 - b. Género: Pop.
 - c. Imagen de portada: Paco selecciona de su equipo el archivo 'canciones.jpg'.
 - d. Licencia: Reconocimiento.
6. Antonio pulsa sobre el botón 'Enviar'.
7. El sistema comunicándose con la base de datos guarda el nuevo álbum, le dice a Antonio que el álbum ha sido creado correctamente y le da la opción de ir a subir canciones.

Escenario 4: Subir canción

Nombre: Subir al álbum 'Mis canciones' la canción 'Mi mundo'.

Descripción: El usuario registrado en el sistema como músico y con el nombre Paco desea subir al álbum 'Mis canciones' la canción 'Mi mundo', para compartir esta canción con el resto de usuarios del sistema.

Actores participantes: Paco y Base de Datos.

Flujo de eventos:

1. El usuario entra en el sistema y se valida de forma correcta como Paco.
2. El sistema se comunica con la base de datos y al comprobar que se trata de un usuario de tipo músico obtiene una lista de los álbumes creados por Paco.
3. Paco escoge el álbum 'Mis canciones' y pulsa en el botón 'Acceder'.
4. El sistema comunicándose con la base de datos obtiene los detalles de este álbum, así como el listado de las canciones que tiene y un pequeño formulario para subir una nueva canción.
5. Paco rellena los campos del formulario para subir canción:
 - a. Título de la canción: Mi mundo.
 - b. Archivo mp3: Paco selecciona de su equipo el archivo 'mi_mundo.mp3'.
6. El sistema comunicándose con la base de datos guarda la nueva canción y le muestra a Antonio un mensaje comunicándole que su canción será subida cuando sea revisada por un administrador.
7. Paco pulsa el botón 'Volver'.
8. El sistema vuelve al paso número 4.

Escenario 5: Editar álbum propio

Nombre: Cambiar de nombre al álbum 'Mis canciones' y la imagen que usa para la portada.

Descripción: El usuario registrado en el sistema como músico y con el nombre Paco desea cambiar el nombre a su álbum 'Mis canciones' por el de 'Mis temas' además de cambiar la actual imagen que usa para la portada de su álbum por no ser de su total agrado.

Actores participantes: Paco y Base de Datos.

Flujo de eventos:

1. El usuario entra en el sistema y se valida de forma correcta como Paco.
2. El sistema se comunica con la base de datos y al comprobar que se trata de un usuario de tipo músico obtiene una lista de los álbumes creados por Paco.
3. Paco escoge el álbum 'Mis canciones' y pulsa en el botón 'Acceder'.
4. El sistema comunicándose con la base de datos obtiene los detalles de este álbum.
5. Paco pulsa el botón 'Editar álbum'.
6. El sistema muestra un formulario para la edición del álbum seleccionado.
7. Paco rellena los siguientes campos:
 - a. Título: Mis temas.
 - b. Género: Pop.
 - c. Imagen de portada: Paco selecciona de su equipo el archivo 'temas.jpg'.
 - d. Licencia: Reconocimiento.
8. Paco pulsa el botón 'Cambiar'.
9. El sistema comunicándose con la base de datos guarda la nueva información del álbum y le muestra a Paco un mensaje de que los cambios se han efectuado correctamente.
10. Paco pulsa el botón 'Volver'.
11. El sistema vuelve al paso 4.

Escenario 6: Eliminar álbum propio

Nombre: Eliminar el álbum 'Mis temas'.

Descripción: El usuario registrado en el sistema como músico y con el nombre Paco no contento con su trabajo decide eliminar el álbum 'Mis temas' con todo su contenido.

Actores participantes: Paco y Base de Datos.

Flujo de eventos:

1. El usuario entra en el sistema y se valida de forma correcta como Paco.
2. El sistema se comunica con la base de datos y al comprobar que se trata de un usuario de tipo músico obtiene una lista de los álbumes creados por Paco.
3. Paco escoge el álbum 'Mis canciones' y pulsa en el botón 'Acceder'.
4. El sistema comunicándose con la base de datos obtiene los detalles de este álbum.
5. Paco pulsa el botón 'Eliminar álbum'.
6. El sistema avisa al usuario de que si continúa eliminará por completo el álbum con su contenido y esta acción no podrá deshacerse.
7. Paco pulsa el botón 'Eliminar'.
8. El sistema comunicándose con la base de datos elimina el álbum 'Mis temas' con las canciones en su interior y vuelve al paso 2.

Escenario 7: Revisar canciones

Nombre: Revisar las canciones subidas al sistema.

Descripción: El usuario registrado en el sistema como administrador_radio y con el nombre de PacoAdmin desea revisar las canciones que han sido subidas en el sistema.

Actores participantes: PacoAdmin y Base de Datos.

Flujo de eventos:

1. El usuario entra en el sistema y se valida de forma correcta como PacoAdmin.
2. El sistema se comunica con la base de datos y como se trata de un usuario administrador_radio obtiene la lista de las canciones que aún están sin revisar.
3. PacoAdmin escucha las 2 canciones que han sido subidas 'Mi mundo' y 'Mi edén' y comprueba que todo está correctamente.
4. PacoAdmin selecciona con un checkbox las dos canciones y pulsa en el botón 'Subir'.
5. El sistema comunicándose con la base de datos añade estas dos canciones a la lista de canciones listas para reproducir y muestra al usuario un mensaje informándole de que por el momento no hay más canciones para revisar.

Escenario 8: Ejecutar algoritmo

Nombre: Ejecutar el algoritmo de filtrado.

Descripción: El usuario registrado en el sistema como administrador_radio y con el nombre PacoAdmin piensa que hace mucho tiempo que no se ejecuta el algoritmo de filtrado y decide hacerlo.

Actores participantes: PacoAdmin y Base de Datos.

Flujo de eventos:

1. El usuario entra en el sistema y se valida de forma correcta como PacoAdmin.
2. El sistema se comunica con la base de datos y como se trata de un usuario administrador_radio obtiene la lista de las canciones que aún están sin revisar.
3. PacoAdmin pulsa en la opción 'Ejecutar algoritmo' en el menú Administrar Radio.
4. El sistema muestra una página diferenciada en dos partes, una parte llamada 'Ejecución automáticamente' y otra 'Ejecución manual'. El usuario por el momento prestará atención a la parte de 'Ejecución manual'.
5. El sistema mostrará en esta parte un mensaje diciendo que la última vez que se ejecuto el algoritmo fue el '01-01-2011 a las 12:30 pm por el usuario 'DavidAdmin'.
6. PacoAdmin decide que ha pasado tiempo suficiente y pulsa el botón 'Ejecutar Algoritmo'.
7. El sistema comunicándose con la base de datos ejecuta el algoritmo de filtrado y actualiza las correspondientes listas de reproducciones y predicciones.
8. El sistema muestra finalmente un mensaje de que el algoritmo se ejecuto satisfactoriamente.

Escenario 9: Eliminar álbum

Nombre: Eliminar el álbum 'Yellow submarine'.

Descripción: El usuario registrado en el sistema como administrador_radio y con el nombre PacoAdmin decide eliminar el álbum 'Yellow submarine' por no ser un álbum que tenga las licencias adecuadas.

Actores participantes: PacoAdmin y Base de Datos.

Flujo de eventos:

1. El usuario entra en el sistema y se valida de forma correcta como PacoAdmin.
2. El sistema se comunica con la base de datos y como se trata de un usuario administrador_radio obtiene la lista de las canciones que aún están sin revisar.
3. PacoAdmin pulsa en la opción 'Eliminar álbum' en el menú Administrar Radio.
4. El sistema se comunica con la base de datos para obtener y mostrar al usuario una lista de los álbumes disponibles en la aplicación.
5. PacoAdmin escoge el álbum 'Yellow submarine' y pulsa el botón 'Eliminar'.
6. El sistema avisa al usuario de que si continúa eliminará por completo el álbum 'Yellow submarine' con su contenido y que esta acción no podrá deshacerse.
7. Paco pulsa el botón 'Eliminar'.
8. El sistema comunicándose con la base de datos elimina el álbum 'Yellow submarine' con las canciones en su interior y vuelve al paso 4.

Escenario 10: Eliminar canciones

Nombre: Eliminar las canciones 'Get up stand up' y 'No woman no cry'.

Descripción: El usuario registrado en el sistema como administrador_radio y con el nombre PacoAdmin decide eliminar las canciones 'Get up stand up' y 'No woman no cry' de álbum 'Bob' por no tener las licencias adecuadas.

Actores participantes: PacoAdmin y Base de Datos.

Flujo de eventos:

1. El usuario entra en el sistema y se valida de forma correcta como PacoAdmin.
2. El sistema se comunica con la base de datos y como se trata de un usuario administrador_radio obtiene la lista de las canciones que aún están sin revisar.
3. PacoAdmin pulsa en la opción 'Eliminar canciones' en el menú Administrar Radio.
4. El sistema se comunica con la base de datos para obtener y mostrar al usuario una lista de los álbumes disponibles en la aplicación.
5. PacoAdmin escoge el álbum 'Bob' y pulsa el botón 'Acceder'.
6. El sistema comunicándose con la base de datos obtiene las canciones pertenecientes al álbum 'Bob'.
7. PacoAdmin escucha las canciones para estar seguro y selecciona ambas canciones acto seguido pulsa el botón 'Eliminar'.
8. El sistema comunicándose con la base de datos elimina las canciones 'Get up stand up' y 'No woman no cry' y muestra un mensaje al usuario para volver a la lista general de álbumes.
9. Paco pulsa el enlace 'Volver'.
10. El sistema vuelve al paso número 4.

Escenario 11: Ver estadísticas

Nombre: Ver las canciones más puntuadas.

Descripción: El usuario registrado en el sistema con el nombre Antonio quiere ver cuáles son las canciones más puntuadas del sistema.

Actores participantes: Antonio y Base de Datos.

Flujo de eventos:

1. El usuario entra en el sistema y se valida de forma correcta como Antonio.
2. El sistema se comunica con la base de datos para obtener las puntuaciones y las predicciones hechas para Antonio y obtiene una lista de reproducción para él.
3. Antonio ignora por el momento su radio y escoge la opción 'Estadísticas' del menú Radio.
4. El sistema se comunica con la base de datos para obtener y mostrar al usuario las cinco canciones más escuchadas del sistema (opción por defecto).
5. Antonio escoge en el apartado seleccionar criterio 'Las más puntuadas' y pulsa el botón 'Aceptar'.
6. El sistema muestra a Antonio las cinco canciones más puntuadas:
7. Antonio decide escuchar la canción más puntuada 'La luz' del álbum 'Haciendo ruido' y pulsa el botón 'Acceder'.
8. El sistema muestra información del álbum concreto 'Haciendo ruido'.

4.4. Diseño del sistema

No cabe duda de que realizar de forma adecuada cada una de las actividades que conlleva la Ingeniería del Software es imprescindible para la realización de un proyecto software de calidad. Por ello, no se puede afirmar que ninguna de estas actividades sea más importante que el resto. Sin embargo, sí podemos decir que la actividad de diseño es la más delicada y la más laboriosa de realizar.

- Es delicada porque, de no llevarse a cabo correctamente, se hace imposible el codificar de manera correcta en la actividad de implementación el modelo obtenido anteriormente en la fase de análisis del sistema. Esto puede repercutir en hacer inútil todo el esfuerzo invertido en las primeras actividades de la Ingeniería del Software.
- Es laboriosa porque las estrategias a seguir para que la traducción entre modelo y código se lleve a cabo correctamente son muy diversas y bastante complejas.

Se puede decir, pues, que el diseño del sistema es la actividad de la Ingeniería del Software en la que se identifican los objetivos finales del sistema y se plantean las diversas estrategias para alcanzarlos en la actividad de implementación.

Sin embargo, el sistema no se suele diseñar de una sola pasada, sino que hemos de diferenciar entre el diseño y estructura de los datos que se van a manejar, y el diseño de la interfaz entre la aplicación y el usuario. Estas dos fases del diseño no se realizan de forma consecutiva una tras otra, sino que lo normal es realizarlas de forma paralela y finalizarlas simultáneamente.

4.4.1. Diseño de los datos

El objetivo de esta fase del diseño software es determinar la estructura que poseen cada uno de los elementos de información del sistema, es decir, la estructura de los datos sobre los que se va a trabajar. Estos elementos son:

- Las **canciones**, de las que sabremos el nombre de la canción, el artista o intérprete de la canción, el álbum al que pertenece, el género musical al que pertenece la canción, la dirección en el servidor en la que se aloja el archivo de sonido asociado a dicha canción, el tipo de licencia, el número de veces que ha sido seleccionada esta canción para ser reproducida y el número de veces que ha sido puntuada.

- Los **usuarios**, de los que conocemos su identificador o nombre de usuario, contraseña, dirección de e-mail, fecha de registro, idioma utilizado, género musical favorito, la dirección en el servidor en la que se aloja una archivo tipo imagen que será utilizada como imagen de perfil o cara virtual y el tipo de usuario (oyente, músico o administrador).

- Las **puntuaciones**, de las que sabemos el usuario que realiza la puntuación, la canción que ha sido puntuada y el valor de dicha puntuación.

- Las **predicciones**, de las que se conocen el usuario sobre el que se ha calculado la predicción, la canción para la que se ha hecho dicha predicción, y el propio valor de la predicción.

- Las **listas de reproducción**, de las que conocemos el usuario cuya lista se tiene almacenada, la canción que forma parte de la lista y un valor numérico que determina su posición dentro de la lista.

- Los **álbumes creados por los músicos**, de las que conocemos el título de este álbum, el tipo de licencia que tiene y el género musical al que pertenece.

- Las **ejecuciones realizadas al algoritmo de filtrado**, vamos a saber la fecha, hora y el usuario que realizó la última ejecución.

Una vez determinados cuales son los elementos de información del sistema, se deben obtener sus representaciones en forma de tablas de una base de datos [24]. Para ello, se debe realizar en primer lugar un diseño conceptual de la base de datos para, posteriormente, obtener las tablas requeridas. Para realizar este diseño conceptual se utilizará el modelo Entidad-Relación (E-R).

4.4.1.1. Esquema conceptual

Necesitamos convertir nuestros elementos de información en entidades o relaciones. En nuestro caso, Canciones y Usuarios pasarán a convertirse en entidades de nuestro esquema conceptual, y Puntuaciones, Predicciones y Listas de reproducción se convierten en relaciones que unen la entidad Canciones con Usuarios. Así, nuestro esquema conceptual quedaría como podemos ver en la Figura 4.5:

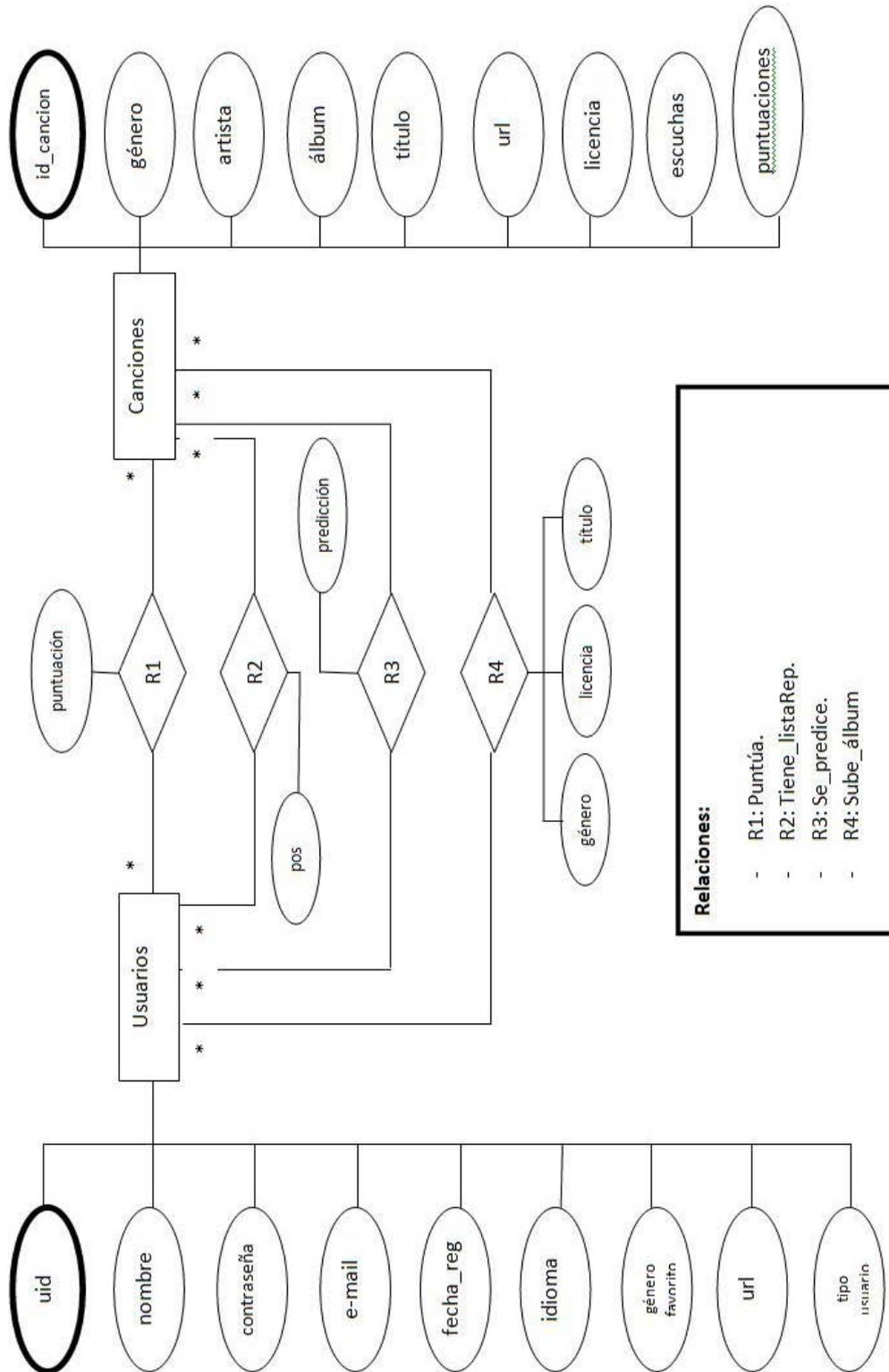


Figura 4.4: Esquema Conceptual de la aplicación

4.4.1.2. Esquema Conceptual Modificado

Para la obtención del Esquema Conceptual Modificado a partir del Esquema Conceptual se deben hacer los cambios que enunciamos a continuación:

- Eliminar todas las entidades débiles.
- Eliminar las relaciones de muchos a muchos.
- Eliminar las relaciones con atributos existentes en el Esquema Conceptual.

En nuestro caso, no es necesaria la eliminación de entidades débiles, ya que no las hay. Realizando las modificaciones oportunas, Obtendríamos cuatro nuevas entidades: Puntuaciones, Predicciones, Listas_rep y Musico_album, derivadas de las cuatro relaciones muchos a muchos existentes en el EC. Nuestro Esquema Conceptual Modificado (ECM) quedará como muestra la Figura 4.6.

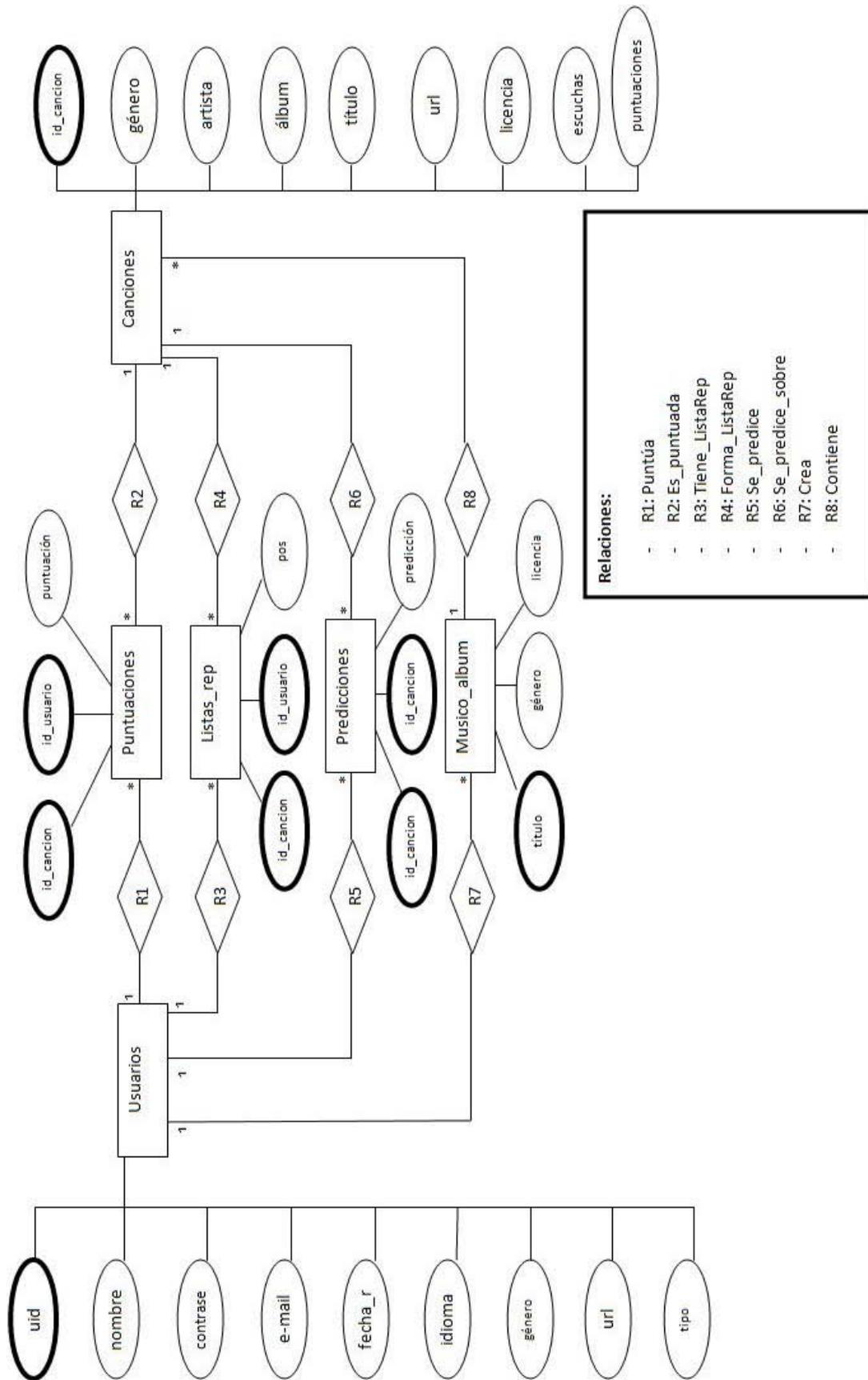


Figura 4.5: Esquema Conceptual Modificado de la aplicación

4.4.1.3. Tablas de la aplicación

Basándonos en el ECM obtenido previamente, podemos determinar un total de 6 tablas en la base de datos, teniendo en cuenta que:

- Cada entidad del ECM se transforma en una tabla.
- Los atributos de una entidad se convierten en los campos de las tablas respectivas.

Por tanto, según el ECM, obtendríamos las siguientes tablas:

- La entidad Canciones da lugar a la tabla CANCIONES.
- La entidad Usuarios pasa a ser la tabla USUARIOS.
- La entidad Puntuaciones se convierte en la tabla PUNTUACIONES.
- La entidad Predicciones da lugar a la tabla PREDICCIONES.
- La entidad Listas_rep se convierte en la tabla LISTAS_REP.
- La entidad Musico_album se convierte en la tabla MUSICO_ALBUM.

Sin embargo, nos permitiremos un par de pequeñas excepciones. Dado que estamos empleando un Gestor de Contenidos (CMS) Drupal para alojar nuestra aplicación Web, ya disponemos a priori de una base de datos con toda la información necesaria para dicho CMS.² En particular, existe una tabla USERS que podemos emplear como la tabla USUARIOS. Además, el atributo 'género favorito', al referirse a información del perfil del usuario, es almacenado en otra tabla especial, denominada PROFILE_VALUES también se almacenarán datos concretos de un usuario tipo músico como el nombre de su 'sitio web', su 'biografía' o el 'género musical' de sus canciones. Asimismo dispondremos de una tabla ROLE donde tendremos los diferentes tipos de roles que pueden adoptar los usuarios y aunque no se ha reflejado en los esquemas conceptuales por no interactuar con las otras tablas, para una mejor administración del algoritmo de filtrado creamos la tabla EJECUCIONES_ALGORITMO donde

² Véase documentación acerca de la base de datos del CMS Drupal para más información.

guardaremos como una bitácora de cuando es ejecutado y por quién y la tabla EJECUCIONES_AUTO donde guardaremos las configuraciones para las ejecuciones automáticas del algoritmo.

Existen muchas más tablas en la base de datos por defecto de Drupal, pero para el desarrollo de nuestro proyecto solamente nos interesan las dos tablas citadas.

De esta forma, nos quedarían las siguientes tablas:

- CANCIONES
- USERS
- PROFILE_VALUES
- PREDICCIONES
- PUNTUACIONES
- LISTAS_REP
- MUSICO_ALBUM
- ROLE
- EJECUCIONES_ALGORITMO
- EJECUCIONES_AUTO

CANCIONES

Con los siguientes atributos cada una:

- ID_CANCION: Entero. Llave primaria. Identificador numérico y unívoco de la canción.
- ARTISTA: Cadena de 40 caracteres. Autor de la canción.
- ÁLBUM: Cadena de 40 caracteres. Nombre del álbum al que pertenece la canción.
- TÍTULO: Cadena de 40 caracteres. Título de la canción.

-URL: Campo de texto. Enlace en el servidor al archivo correspondiente a la canción.

-GÉNERO: Conjunto de valores. Género al que la canción pertenece. Puede tomar los siguientes valores (acústica, rock, techno, instrumental, pop, house, psicodelic, ambiental o bso).

-LICENCIA: Conjunto de valores. Licencia al que la canción pertenece. Puede tomar los siguientes valores (reconocimiento, sin_obra, sin_obra_no_comercial, no_comercial, no_comercial_compartir_igual o compartir_igual).

-ESCUCHAS: Entero de valor positivo y por defecto 0. Este valor nos dirá el número de veces que la canción es seleccionada para reproducciones en la radio.

-BORRADO: Cadena de 2 caracteres. El valor se utilizará para hacer un borrado lógico, es decir, 'si' si la canción se encuentra borrada y 'no' (valor por defecto) en caso contrario.

-REVISADA: Cadena de 2 caracteres. El valor se utilizará para que una canción tenga que ser aceptada o revisada por un usuario administrador antes de que pueda ser reproducida en la radio. Los valores pueden ser 'no' (valor por defecto) la canción aun no ha sido revisada y 'si' en casa contrario.

-PUNTUACIONES: Entero de valor positivo y por defecto 0. Con este dato conoceremos el número de veces que ha sido puntuada la canción.

	id_cancion	genero	artista	album	url	titulo	licencia	escuchas	borrado	revisada	puntuaciones
<input type="checkbox"/>	195	ambiental	Saregama	Samadhi	musica/Saregama-Samadhi/07 - Seven Eleven.mp3	Seven eleven		4	no	si	7
<input type="checkbox"/>	196	ambiental	Saregama	Samadhi	musica/Saregama-Samadhi/08 - Desert Sunset.mp3	Desert Sunset		6	no	si	8
<input type="checkbox"/>	197	ambiental	Saregama	Samadhi	musica/Saregama-Samadhi/09 - Ruins of Bethel.mp3	Ruins of Bethel		0	no	si	5
<input type="checkbox"/>	202	rock	Petete	Haciendo ruido	musica/Petete-Haciendo ruido/Todo va bien.mp3	Todo va bien	sin_obra_no_comercial	38	no	si	3
<input type="checkbox"/>	203	rock	Petete	Haciendo ruido	musica/Petete-Haciendo ruido/La cancion contra los...	La cancion contra los jefes	sin_obra_no_comercial	52	no	si	3
<input type="checkbox"/>	204	rock	Petete	Haciendo ruido	musica/Petete-Haciendo ruido/La luz.mp3	La luz	sin_obra_no_comercial	49	no	si	3
<input type="checkbox"/>	205	rock	Petete	Haciendo ruido	musica/Petete-Haciendo ruido/Septimo arte.mp3	Septimo arte	sin_obra_no_comercial	58	no	si	1
<input type="checkbox"/>	206	rock	Petete	Haciendo ruido	musica/Petete-Haciendo ruido/Vivir adrede.mp3	Vivir adrede	sin_obra_no_comercial	64	no	si	2

Figura 4.6: Ejemplo de tuplas en la tabla CANCIONES

USUARIOS

Se trata de una tabla que contiene tantas filas como usuarios hay registrados en el portal Web de la radio.

La tabla 'users' guarda un amplio conjunto de atributos, los que realmente interesan para llevar a cabo la gestión de puntuaciones sobre canciones son:

- UID: Entero. Llave primaria. Identificador numérico único para el usuario.

- Existe un usuario especial, el usuario anónimo, cuyo UID es 0. Este dato es útil para condicionar ciertos aspectos del sitio Web según el usuario esté registrado o sólo sea un invitado.
- NAME: Cadena de 60 caracteres. Nombre del usuario.
- PASS: Cadena de 32 caracteres. Contraseña del usuario codificada mediante un algoritmo de cifrado.
- MAIL: Cadena de 64 caracteres. Dirección de correo-e del usuario.
- LANGUAGE: Cadena de 12 caracteres. Código del idioma empleado por el usuario.
- ACCESS: Entero. Tiempo transcurrido desde el último acceso del usuario a la aplicación.
- PICTURE: Campo de texto. Enlace en el servidor al archivo correspondiente a la foto utilizada como foto de perfil.

	uid	name	pass	mail	language	access	picture
<input type="checkbox"/>	0					0	
<input type="checkbox"/>	1	ivanp	898cd2802f73199f1b2ea19be9844fa	ivanp@ujaen.es	es	1293449826	sites/default/files/pictures/picture-1.jpg
<input type="checkbox"/>	8	atmjesus	cbfecc1a82915574e88db83ff02cd5a2	atmjesus@gmail.com	es	1241035368	
<input type="checkbox"/>	5	ivanp2	1f492b64282241824ffb530ad871955	ivanpc86@gmail.com	es	1262174702	
<input type="checkbox"/>	6	martin	847eededf3dc9daea212fe0fc9f7252d	martin@ujaen.es	es	1269446080	
<input type="checkbox"/>	7	mestevez	01a99e6c55e5b70fc316bbe783eb97c8	mestevez@ujaen.es		1241253508	

Figura 4.7: Ejemplo de tuplas en la tabla USUARIOS

PUNTUACIONES

Tabla que contiene los siguientes campos:

- ID_USUARIO: Entero. Llave primaria. Llave externa. Identificador del usuario.
- ID_CANCION: Entero. Llave primaria. Llave externa. Identificador de la canción.
- PUNTUACIÓN: Entero de longitud 1. Puntuación (1, 2, 3, 4 ó 5) del usuario sobre la canción.

	id_usuario	id_cancion	puntuacion
<input type="checkbox"/>	1	189	3
<input type="checkbox"/>	7	19	1
<input type="checkbox"/>	1	124	2
<input type="checkbox"/>	5	25	3
<input type="checkbox"/>	1	10	3
<input type="checkbox"/>	1	31	2
<input type="checkbox"/>	1	156	4
<input type="checkbox"/>	8	142	1
<input type="checkbox"/>	8	143	2
<input type="checkbox"/>	8	144	3
<input type="checkbox"/>	8	145	4
<input type="checkbox"/>	8	146	5
<input type="checkbox"/>	8	147	1
<input type="checkbox"/>	8	148	2
<input type="checkbox"/>	8	149	4

Figura 4.8: Ejemplo de tuplas en la tabla PUNTUACIONES

PREDICCIONES

Tabla que contiene los siguientes campos:

- ID_USUARIO: Entero. Llave primaria. Llave externa. Identificador del usuario.
- ID_CANCION: Entero. Llave primaria. Llave externa. Identificador de la canción.
- PREDICCIÓN: Real. Valor entre 0 y 5.25. Predicción para el usuario sobre la canción.

←T→	id_usuario	id_cancion	prediccion
<input type="checkbox"/>  	120	193	0
<input type="checkbox"/>  	120	194	0
<input type="checkbox"/>  	120	195	0
<input type="checkbox"/>  	120	196	0
<input type="checkbox"/>  	120	197	0
<input type="checkbox"/>  	120	202	5
<input type="checkbox"/>  	120	203	4.666666666666667
<input type="checkbox"/>  	120	204	5

Figura 4.9: Ejemplo de tuplas en la tabla PREDICCIONES

LISTAS_REP

Tabla que contiene los siguientes campos:

- ID_USUARIO: Entero. Llave primaria. Llave externa. Identificador del usuario.
- ID_CANCION: Entero. Llave primaria. Llave externa. Identificador de la canción.
- POS: Real. Valor que determina la posición de la canción en la lista de reproducción del usuario.

←T→	id_usuario	id_cancion	pos
<input type="checkbox"/>  	1	24	6
<input type="checkbox"/>  	1	133	4
<input type="checkbox"/>  	1	22	0
<input type="checkbox"/>  	1	140	7
<input type="checkbox"/>  	1	125	1
<input type="checkbox"/>  	1	7	3
<input type="checkbox"/>  	99	153	0

Figura 4.10: Ejemplo de tuplas en la tabla PREDICCIONES

PROFILE_VALUES³

- FID: Entero. Llave primaria. Identificador del campo de perfil de usuario.
- UID: Entero. Llave primaria. Llave externa. Identificador del usuario.
- VALUE: Conjunto de valores. Género favorito del usuario. Puede tomar los valores siguientes (acústica, rock, techno, instrumental, pop, house, psicodelic, ambiental o bso).

			fid	uid	value
<input type="checkbox"/>			2	8	pop
<input type="checkbox"/>			2	9	rock
<input type="checkbox"/>			2	10	rock
<input type="checkbox"/>			2	5	acustica
<input type="checkbox"/>			2	13	acustica
<input type="checkbox"/>			2	14	rock
<input type="checkbox"/>			2	1	acustica

Figura 4.11: Ejemplo de tuplas en la tabla PROFILE_VALUES

MUSICO_ALBUM

- TITULO: Cadena de 40 caracteres. Clave primaria. Título del álbum que deberá ser único.
- ID_MUSICO: Entero. Llave externa. Identificador del usuario de tipo músico.
- GÉNERO: Conjunto de valores. Género al que el álbum pertenece. Puede tomar los siguientes valores (acústica, rock, techno, instrumental, pop, house, psicodelic, ambiental o bso).
- LICENCIA: Conjunto de valores. Licencia al que el álbum pertenece. Puede tomar los siguientes valores (reconocimiento, sin_obra, sin_obra_no_comercial, no_comercial, no_comercial_compartir_igual o compartir_igual).
- BORRADO: Cadena de 2 caracteres. El valor se utilizará para hacer un borrado lógico, es decir, 'sí' si el álbum se encuentra borrado y 'no' (valor por defecto) en caso contrario.

³ Esta tabla es generada por el sistema de gestión de contenidos para almacenar valores de campos de perfil para los usuarios. En nuestro diseño de datos original, formaría parte de la tabla Usuarios.

←T→		titulo	id_musico	genero	licencia	borrado
<input type="checkbox"/>	 	Haciendo ruido	104	rock	sin_obra_no_comercial	no

Figura 4.12: Ejemplo de tuplas en la tabla MUSICO_ALBUM

ROLE

- RID: Entero. Llave primaria. Identificador del campo del rol.
- NAME: Cadena de caracteres. Nombre del rol.

←T→		rid	name
<input type="checkbox"/>	 	1	anonymous user
<input type="checkbox"/>	 	2	authenticated user
<input type="checkbox"/>	 	3	Administrador
<input type="checkbox"/>	 	4	musico
<input type="checkbox"/>	 	5	oyente
<input type="checkbox"/>	 	6	Administrador_radio

Figura 4.13: Ejemplo de tuplas en la tabla ROLE

EJECUCIONES_ALGORITMO

- ID: Entero. Llave primaria. Identificador numérico y unívoco de la tupla.
- FECHA: Cadena de caracteres. La fecha donde se ejecuto el algoritmo con el formato día-mes-año.
- HORA: Cadena de caracteres. La hora a la cual se ejecuto el algoritmo con el formato hora:minutos:segundos pm/am.
- USUARIO: Cadena de caracteres. El nombre del usuario de tipo administrador que ejecuto el algoritmo.

←T→		id	fecha	hora	usuario
<input type="checkbox"/>	 	54	15-12-2010	06:37:42 pm	ivanp
<input type="checkbox"/>	 	55	15-12-2010	08:50:32 pm	dparras
<input type="checkbox"/>	 	56	15-12-2010	09:01:14 pm	dparras
<input type="checkbox"/>	 	57	16-12-2010	12:34:32 am	dparras
<input type="checkbox"/>	 	58	16-12-2010	02:04:17 pm	dparras
<input type="checkbox"/>	 	59	16-12-2010	02:54:25 pm	dparras
<input type="checkbox"/>	 	60	17-12-2010	10:16:43 am	macaAdmin
<input type="checkbox"/>	 	61	17-12-2010	08:47:35 pm	dparras
<input type="checkbox"/>	 	62	20-12-2010	04:54:09 pm	dparras
<input type="checkbox"/>	 	63	21-12-2010	01:11:09 pm	dparras
<input type="checkbox"/>	 	64	21-12-2010	01:11:09 pm	dparras
<input type="checkbox"/>	 	65	22-12-2010	12:06:44 am	dparras

Figura 4.14: Ejemplo de tuplas en la tabla EJECUCIONES_ALGORITMO

EJECUCIONES_AUTO

- ID: Entero. Llave primaria. Identificador numérico y unívoco de la tupla.
- MINUTO: Entero positivo. Nos indicará el minuto donde estará programada la ejecución.
- HORA: Entero positivo. Indica la hora a la cual se ejecutará el algoritmo automáticamente.
- DIA: Entero. Nos indica el día que queremos que se ejecute el algoritmo. Un 0 para el domingo, un 1 para el lunes,..., hasta un 6 para el sábado. En caso de que el valor sea -1 indicará que el algoritmo se ejecutará diariamente.

←T→		id	minuto	hora	día	
<input type="checkbox"/>			1	30	13	-1
<input type="checkbox"/>			2	30	20	-1
<input type="checkbox"/>			3	28	21	1
<input type="checkbox"/>			4	24	21	6
<input type="checkbox"/>			5	25	6	6

- Figura 4.15: Ejemplo de tuplas en la tabla EJECUCIONES_AUTO

4.4.2. Diseño de la Interfaz

En esta etapa del diseño del sistema software se define cual va a ser la apariencia visual de la aplicación, es decir, se define la interfaz visual entre el usuario y la aplicación. Sin duda, llevar a cabo un buen diseño de la interfaz resulta decisivo debido a que esta debe ser atractiva para el usuario de la aplicación pero al mismo tiempo debe resultar fácil de entender y de trabajar sobre ella [19].

Esta importancia se acrecienta aun más en el caso de nuestro proyecto ya que la interfaz de nuestro proyecto es una interfaz Web y la usabilidad Web es un tema candente, sujeto a cierta controversia [13]. Esto es así porque para las aplicaciones con interfaces Web no existe una guía de estilo estándar como existe, por ejemplo, para el desarrollo de interfaces para aplicaciones de escritorio en sistemas operativos como Windows XP/Vista/7 y que resulten atractivas y familiares al mismo tiempo. Cada

programador, desarrollador o diseñador Web debe definir, pues, su propia guía de estilo y procurar que, en base a ella, la interfaz resultante alcance un nivel digno de atractivo visual, familiaridad y facilidad de uso.

En este apartado definiremos nuestra guía de estilo de la aplicación, describiremos y analizaremos las metáforas y revisaremos el diseño de pantallas y los caminos de navegación (storyboards) empleados para el desarrollo de la interfaz de nuestro prototipo de aplicación software.

4.4.2.1. Guía de Estilo

Antes de entrar de lleno en el diseño de la interfaz de usuario, es conveniente definir el estilo de la misma. Esto es de suma importancia cuando el diseño va a ser compartido entre varios diseñadores, ya que ayuda a mantener la coherencia interna de la interfaz.

Sin embargo, a pesar de lo que pueda parecer en un principio, también es de gran utilidad definir una guía de estilo cuando sólo hay un diseñador encargado de la interfaz. Esto es debido a varias razones:

- En ocasiones es posible que mantener la coherencia y consistencia de una interfaz, si esta es muy grande o muy ambiciosa, sea complicado incluso si sólo hay un diseñador.
- El diseñador primitivo puede, por las más diversas razones, abandonar el diseño. Por tanto, es de gran utilidad para sus sustitutos contar con una guía de estilo predefinida para no tener que empezar desde cero de nuevo. Esto es también aplicable cuando no es el diseñador original el que se encarga de la actualización o el mantenimiento de la interfaz.

Quedando demostrada la utilidad del uso de guías de estilo, pasamos a definir las reglas y normas que contendrá la guía de estilo de nuestra interfaz:

Fuentes

Estándar

- Tipo: Lucida Grande.
- Color: #000000
- Tamaño: 1.125 em.

Estándar (nombres de campo).

- Tipo: Lucida Sans Unicode.
- Color: #000000
- Tamaño: 1.125 em.

Menús

- Tipo: Arial
- Color: #0000FF
- Tamaño: 1.0 em.

Información listas de reproducción

- Tipo: Lucida Grande
- Color: #000000
- Tamaño: 1.125 em.

Cabeceras

- Tipo: Times New Roman
- Color: #005a8c
- Tamaño: 1.1em.

Enlaces

Los enlaces aparecerán del mismo color que el empleado en la fuente en los menús. El texto no estará subrayado, sino que al pasar el cursor por encima el texto será rodeado de un color azul más intenso; sin embargo las imágenes o el texto de título de la página simplemente serán subrayados. El destino de los enlaces será por norma general una pestaña diferente. Para los enlaces de elementos en los diferentes menús del sitio, el enlace activo cambiará de su color azul a un color negro.

Colores

El color predominante donde se encuentra la mayoría del texto será el blanco marfil, el color de las cajas de menús será azul en la parte superior difuminándose hacia blanco de forma descendente, para el resto del sitio un azul más oscuro y la cabecera del sitio está formada por una imagen que recuerda al dial de una radio real.

Alineación

Texto y contenido alineado a la izquierda, salvo tablas cuya alineación es centrada.

4.4.2.2. Metáforas

Una metáfora es el empleo de un objeto con un significado o dentro de un contexto diferente al habitual. En el diseño de una interfaz gráfica, la utilización de metáforas resulta muy útil ya que permiten al usuario, mediante la comparación con otro objeto o concepto, comprender de forma más intuitiva las diversas tareas que la interfaz permite desarrollar. Las metáforas juegan un papel muy importante en el éxito o fracaso de una interfaz, por tanto deben diseñarse con mucho cuidado.

Al igual que pasa en el ámbito de la literatura, para que una metáfora cumpla con su cometido, el desarrollador de la aplicación y el usuario final de la misma deben tener una base social y cultural similar. Es muy posible que el uso de un icono de manera metafórica sea entendido de una manera por el usuario occidental y de otra bien distinta por un usuario oriental. Pensemos, por ejemplo, en el icono de una mano con el pulgar levantado: para un usuario occidental es muy probable que signifique

que la tarea se ha realizado con éxito. Sin embargo, un usuario natural de Turquía, por ejemplo, consideraría este gesto como un acto ofensivo.

Es necesario, pues, tener en cuenta la base cultural de los usuarios para diseñar una buena metáfora. En definitiva, hay que intentar que las metáforas empleadas sean lo más universales posibles, a fin de que sean comprendidas a la perfección por la mayor parte del público potencial.

Las aplicaciones de escritorio de Windows actuales suelen seguir la Guía de Estilo Vista y utilizan una serie de metáforas con las que el usuario está plenamente familiarizado (por ejemplo, una papelera con el símbolo de la triple flecha de reciclado indica el lugar al que provisionalmente se destinan aquellos archivos que han sido borrados). En el mundo de las aplicaciones Web también existe una gran cantidad de metáforas de amplia difusión como, por ejemplo, el conocido icono del carrito de la compra que emplean casi todos los comercios online.

Pero las metáforas no solo dependen del tipo de aplicación (escritorio o Web) sino también del ámbito de la misma. Por ejemplo, el carrito de la compra es una metáfora conocida por todos, pero si nuestra aplicación no va a vender producto alguno al usuario no conviene utilizarla ya que puede dar lugar a confusión.

En nuestra aplicación hemos utilizado las siguientes metáforas:

-  Esta metáfora aparece como la primera opción en los paneles de puntuaciones, representa la puntuación de menor valor, 1 o puntuación 'Mala', como vemos al tener solo una estrella de cinco posibles representará el mínimo valor.
-  Esta metáfora es la segunda opción en los paneles de puntuación, representa valor 2 o puntuación 'Regular', aquí podemos observar que tenemos dos estrellas de cinco posibles.
-  Esta metáfora es la tercera opción en los paneles de puntuación, representa valor 3 o puntuación 'Buena', aquí podemos observar que tenemos tres estrellas de cinco posibles.

-  Esta metáfora es la cuarta opción en los paneles de puntuación, representa valor 4 o puntuación 'Muy Buena', en este caso tendremos cuatro de cinco estrellas posibles.
-  Esta metáfora es la mejor opción en los paneles de puntuación, representa valor 5 o puntuación 'Excelente', en este caso tendremos todas las estrellas posibles por eso significa el máximo valor.
- Para entender mejor el funcionamiento de estas metáforas de estrella cuando pasemos el cursor por encima de ellas se nos informará del tipo de puntuación de cada una (mala, regular, buena, muy buena, o excelente), veamos el diseño del formulario para puntuar canciones:



Figura 4.16: Panel de puntuaciones de la aplicación.



- El propio panel de controles del reproductor de la radio es una metáfora de los radios, equipos de música y demás aparatos de reproducción de audio y vídeo en general. Incluye botones para pasar a la anterior y siguiente canción, comenzar, detener o reanudar la reproducción, ajustar el volumen y desplazarse dentro de la canción que está siendo reproducida en el momento.
- La disposición de los diferentes álbumes, de forma matricial, en la sección "álbumes" recuerda a cómo los discos de música son colocados en un escaparate real.

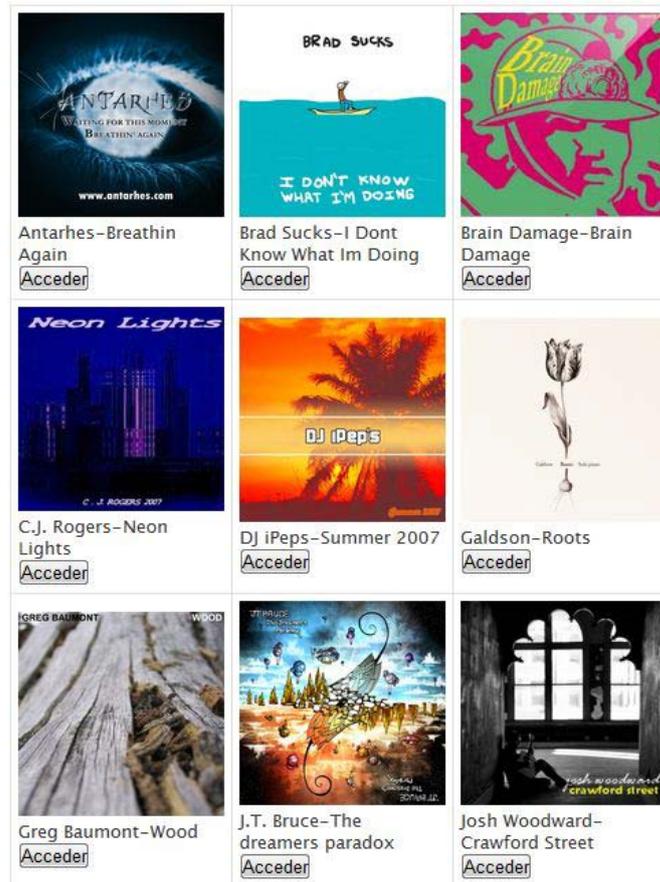


Figura 4.17: Disposición de la lista de álbumes.

- La disposición de los músicos en nuestro sitio intenta asimilarse a la manera en que lo hacen las listas de amigos en las redes sociales como facebook.

Músicos



Figura 4.18: Disposición de la lista de músicos.

-  Esta metáfora aparece en el libro de visitas e indica al usuario dónde debe dirigirse para escribir o responder un comentario.
-  Esta metáfora aparece en el libro de visitas e indica al usuario a dónde debe dirigirse para editar un comentario anteriormente publicado por él.
-  Esta metáfora aparece en el libro de visitas e indica al usuario a dónde debe dirigirse para eliminar un comentario anteriormente escrito por él.
-  Metáfora que representa un usuario registrado en la página.
-  Metáfora útil para recomendar nuestro sitio en la famosa red social facebook (se explicará con todo lujo de detalles en el Anexo A: Manual de usuario).
-  Metáfora para publicar un enlace de nuestro sitio en la famosa red social twitter (se explicará con todo lujo de detalles en el Anexo A: Manual de usuario).

4.4.2.3. Pantallas de la aplicación

En este apartado, vamos a definir la estructura de nuestra interfaz con el usuario.

Como explicamos anteriormente tendremos diferentes tipos de usuarios en nuestra aplicación, cada uno ellos tendrá una funcionalidad diferente del resto, por tanto intentaremos hacer una interfaz personalizada dependiendo del tipo de usuario que interactúe con el sistema.

Así que, a continuación explicaremos las diferentes pantallas dependiendo del tipo de usuario y la diferenciación de las partes de radio online, menú de usuario, administrador de radio y músico.

Pantalla principal/radio (Usuario anónimo).

Como podemos ver el usuario anónimo podrá optar a las funciones de radio, pero no podrá utilizar las funciones de menú de usuario, en su lugar tendremos un pequeño formulario para que el usuario inicie sesión si lo desea.

The screenshot shows the OL-Radio website interface. At the top, there's a header with 'OL-Radio' and a background image of a radio dial. Below the header, there's a navigation menu on the left with options like 'Lanzar radio', 'Álbumes', 'Músicos', 'Estadísticas', 'Libro de visitas', and 'Quiénes somos'. A red circle highlights this menu. In the center, there's a login form with fields for 'Usuario:' and 'Contraseña:', and a button 'Iniciar sesión'. To the right, there's a section for 'ENLACES' with a 'SINBAD²' logo and a 'QUIÉN ESTÁ EN LÍNEA' section showing 'Actualmente hay 0 usuarios y 1 invitado en línea.'. The main content area features a 'Radio Colaborativa bajo Licencia Creative Commons' section with a text prompt: 'Esta es la **Radio General**, para poder recomendar canciones regístrate por favor.' Below this is a music player with a playlist of 5 songs: 1 - Brain Damage - Reves de metal, 2 - Tryad - Beat Into Submission, 3 - Lonah - Artiste, 4 - Petete - La luz, 5 - Josh Woodward - Afterglow. A red arrow points from a text box to the navigation menu. The text box contains the following text:

Como vemos el usuario anónimo tendrá acceso a las acciones de radio pero no a las opciones de usuario, en su lugar tendrá el formulario de registro.

Figura 4.19: Pantalla principal/radio para el usuario anónimo

Pantalla principal/radio (Usuario registrado de tipo oyente)

En este caso el usuario de tipo oyente podrá optar a las opciones de usuario como mensajes, mi cuenta o mis puntuaciones. La pantalla principal seguirá siendo la radio pero esta vez será personalizada con las recomendaciones que el sistema le ofrece al usuario.

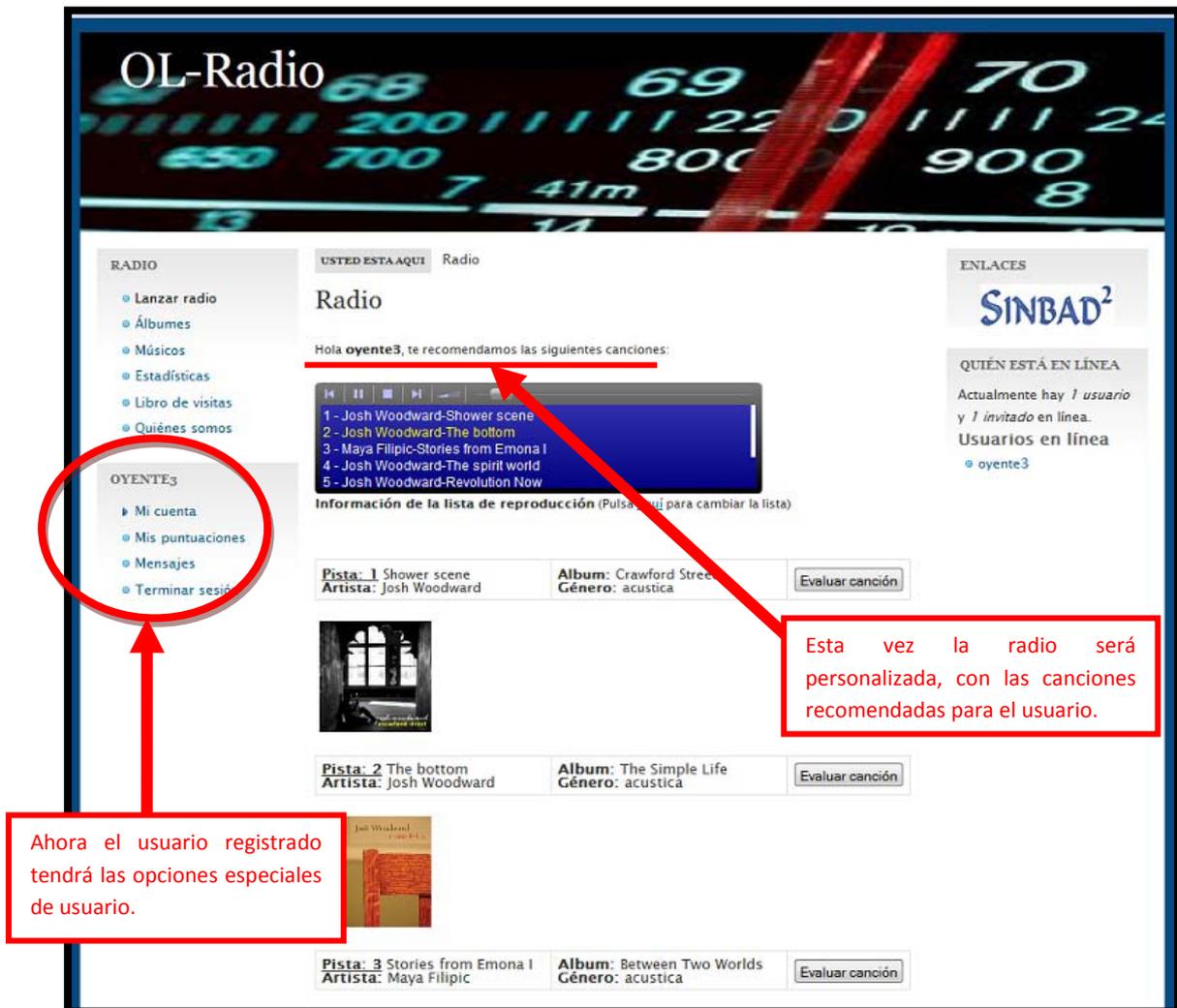


Figura 4.20: Pantalla principal/radio para el usuario registrado de tipo oyente

Pantalla principal/Mis álbumes (Usuario registrado de tipo músico)

El usuario de tipo músico tendrá la característica especial en nuestro sitio de poder crear álbumes y subir su propia música. Por tanto la pantalla principal de un usuario de tipo músico será la lista de sus álbumes para que empiece a trabajar sobre ellos (subir canciones, editarlos, etc.). Además de tener acceso al menú 'Mi música'.

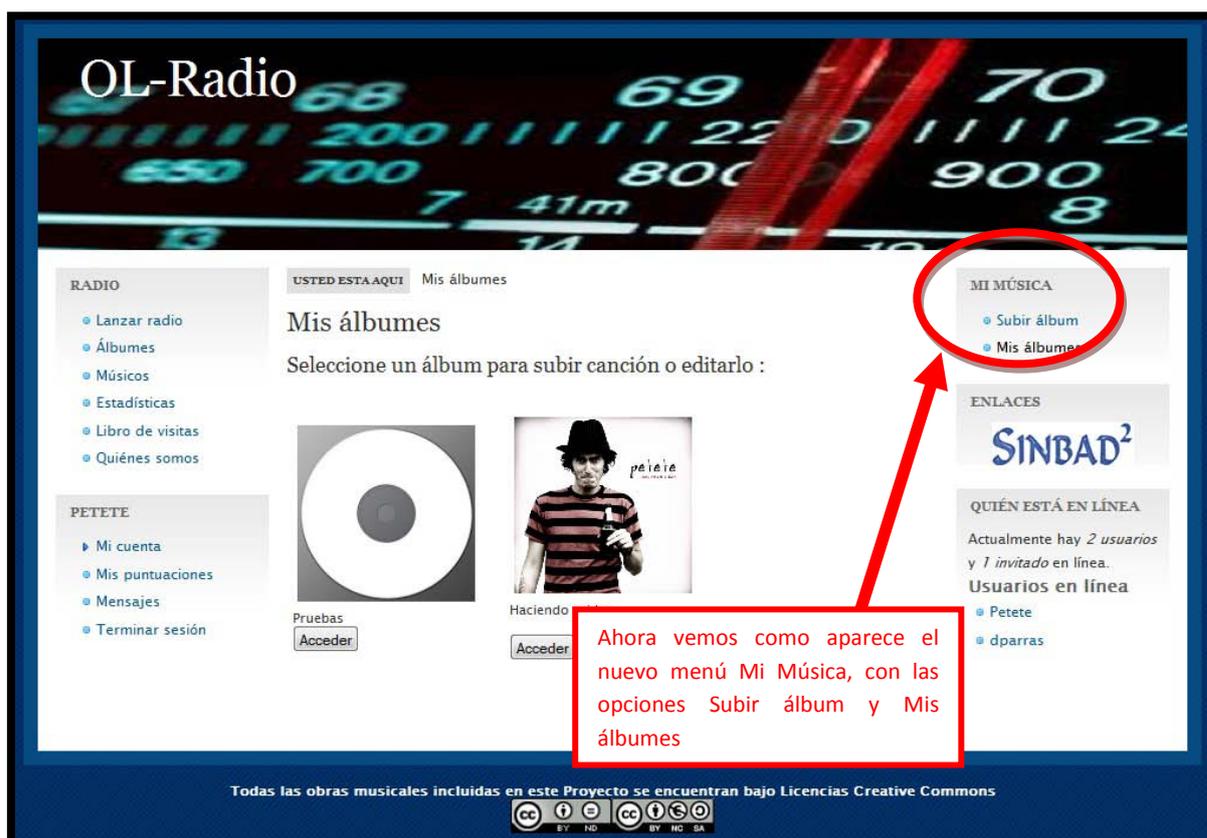
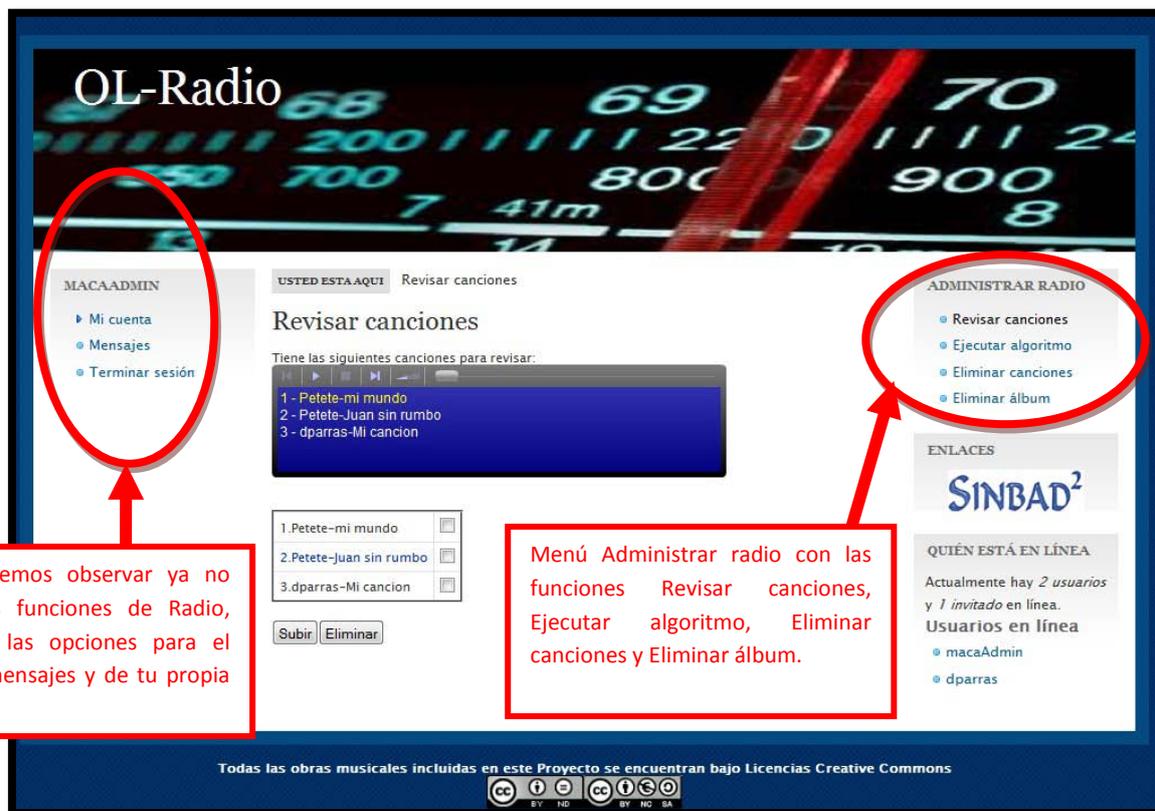


Figura 4.21: Pantalla principal/radio para el usuario registrado de tipo músico

Pantalla principal/Revisar canciones (Usuario registrado de tipo administrador radio)

El usuario de tipo administrador radio no optará a las funciones de radio ya que su función es básicamente la de administrar la radio, por tanto tendrá un menú con las funciones de administración. Referente a la parte de menú específico de usuario solamente tendrá acceso las funciones para envío de mensajes para que el administrador se pueda comunicar con los usuarios para dar avisos de eliminaciones por ejemplo.



Como podemos observar ya no existen las funciones de Radio, solamente las opciones para el envío de mensajes y de tu propia cuenta.

Menú Administrar radio con las funciones Revisar canciones, Ejecutar algoritmo, Eliminar canciones y Eliminar álbum.

Figura 4.22: Pantalla principal/radio para el usuario registrado de tipo administrador radio

Pantalla Músicos

Pantalla con la lista de los músicos colaboradores con el sitio.



Figura 4.23: Músicos

Pantalla Subir álbum (Usuario registrado de tipo músico)

Como podemos ver a través de este formulario un usuario puede subir o crear un nuevo álbum.

USTED ESTÁ AQUÍ Subir álbum

Subir álbum

Dime el título de tu álbum :

Seleccione el género que mas se adapte a tu álbum :

- Rock
- Acústica
- Techno
- Instrumental
- Pop
- House
- Psicodelic
- Ambiental

Sube una imagen para la portada de tu álbum :

No se h...archivo

El archivo debe ser de tipo Imagen(jpg,jpeg,png,gif,bmp) y tener un tamaño máximo de 5 MB.

Seleccione la licencia oportuna :

<input checked="" type="radio"/>	Reconocimiento: Tu obra puede ser distribuida, copiada y exhibida, dejando claro que tú eres el autor.
<input type="radio"/>	Reconocimiento - Sin obra derivada: Tu obra puede ser distribuida, copiada y exhibida, pero no pueden crear otras obras a partir de la original.
<input type="radio"/>	Reconocimiento - Sin obra derivada - No comercial: Tu obra puede ser distribuida, copiada y exhibida, pero no pueden obtener ningún beneficio económico ni realizar otras obras a partir de la original.
<input type="radio"/>	Reconocimiento - No comercial: Tu obra puede ser distribuida, copiada y exhibida, pero no pueden obtener ningún beneficio económico.
<input type="radio"/>	Reconocimiento - No comercial - Compartir igual: Tu obra puede ser distribuida, copiada y exhibida, pero no pueden obtener ningún beneficio económico. Pueden crear otras obras a partir de la tuya siempre que lleven la misma licencia que tu obra original.
<input type="radio"/>	Reconocimiento - Compartir igual: Tu obra puede ser distribuida, copiada y exhibida. Pueden crear otras obras a partir de la tuya siempre que lleven la misma licencia que tu obra original.

Figura 4.24: Pantalla Subir Álbum

Pantalla Editar álbum (Usuario registrado de tipo músico)

A través de aquí un usuario de tipo músico podrá subir canciones, eliminar canciones y editar o eliminar su propio álbum.

The screenshot displays the 'Edit Album' interface for a user named 'petete'. The page is divided into several sections:

- Left Sidebar:** Contains navigation links for 'RADIO' (Lanzar radio, Álbumes, Músicos, Estadísticas, Libro de visitas, Quiénes somos) and 'PETETE' (Mi cuenta, Mis puntuaciones, Mensajes, Terminar sesión).
- Top Center:** Shows 'USTED ESTA AQUÍ Mis álbumes' and the title 'Mis álbumes'.
- Album Selection:** A section titled 'Seleccione un álbum para subir canción o editarlo:' featuring two album covers: 'Pruebas' and 'Haciendo ruido'. Each cover has an 'Acceder' button.
- Album Options:** A section titled 'Opciones del álbum Haciendo ruido' showing album details:
 - Artista: Petete
 - Album: Haciendo ruido
 - Género: rock
 - Número canciones: 5
 - Tipo de licencia: Reconocimiento - Sin obra derivada - No comercial
 Below the details are 'Eliminar Album' and 'Editar Album' buttons.
- Player:** A music player interface showing a list of songs:
 - 1 - Petete- Todo va bien
 - 2 - Petete-La cancion contra los jefes
 - 3 - Petete-La luz
- Songs List:** A table listing all five songs with 'Eliminar' buttons for each:

1. Petete- Todo va bien	Eliminar
2. Petete- La cancion contra los jefes	Eliminar
3. Petete- La luz	Eliminar
4. Petete- Septimo arte	Eliminar
5. Petete- Vivir adrede	Eliminar
- Upload Form:** A section titled 'Sube una canción:' with a form to 'Dime el título de tu canción', a 'Sube el archivo mp3' button (with a 'Seleccionar archivo' link), and an 'Enviar' button. A note specifies: 'El archivo tiene que ser tipo mp3 y tener un tamaño máximo de 10MB.'
- Right Sidebar:** Contains 'MI MÚSICA' (Subir álbum, Mis álbumes), 'ENLACES' (SINBAD²), and 'QUIÉN ESTÁ EN LÍNEA' (Actualmente hay 1 usuario y 1 invitado en línea, Usuarios en línea: Petete).

Figura 4.25: Pantalla Editar Álbum

Pantalla Ejecutar algoritmo (Usuario registrado de administrador radio)

El usuario de tipo administrador radio podrá ejecutar manualmente si lo desea el algoritmo de filtrado o configurar la ejecución automática.

OL-Radio

MACAADMIN

- Mi cuenta
- Mensajes
- Terminar sesión

USTED ESTA AQUÍ Ejecutar algoritmo

Ejecutar algoritmo

Ejecución automática

El algoritmo en estos momentos se ejecuta todos los sábados a las 6:25.

Si desea cambiarlo, rellene los siguientes campos según sus preferencias.

Hora: Minuto: Día:

Ejecución manual

Si deseas ejecutarlo manualmente pulse el botón (esta acción puede tomar varios minutos).

Última ejecución del algoritmo el **6-2-2011** a las **06:47:02 am** de forma **automática**.

ADMINISTRAR RADIO

- Revisar canciones
- Ejecutar algoritmo
- Eliminar canciones
- Eliminar álbum

ENLACES

SINBAD²

QUIÉN ESTÁ EN LÍNEA

Actualmente hay **1** usuario y **0** invitados en línea.

Usuarios en línea

- macaAdmin

Todas las obras musicales incluidas en este Proyecto se encuentran bajo Licencias Creative Commons

Figura 4.26: Pantalla Ejecutar algoritmo

Pantallas Eliminar canciones y Eliminar álbum (Usuario registrado de administrador radio)

Por ultimo vemos las dos pantallas por las que un usuario puede eliminar tanto álbumes como canciones.

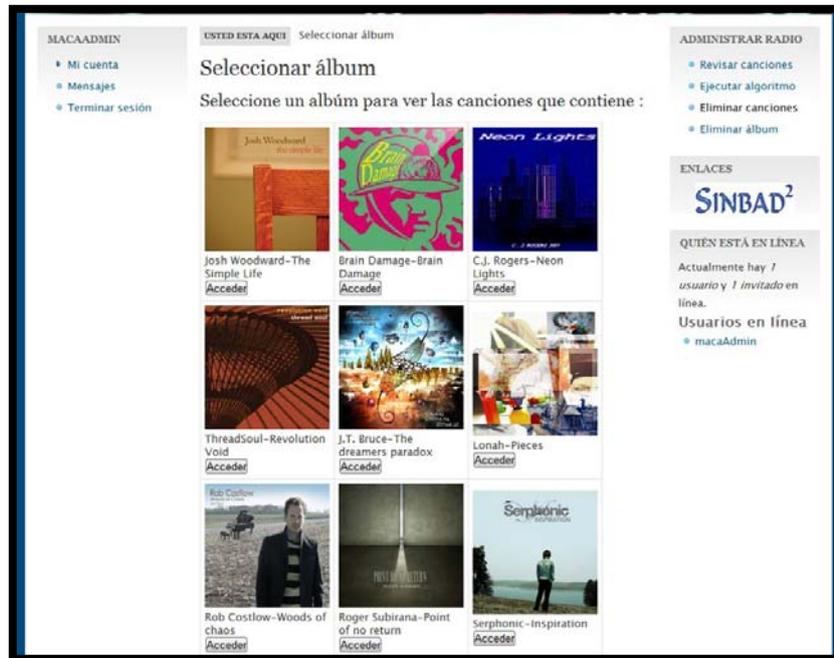


Figura 4.27: Pantalla Eliminar canciones.

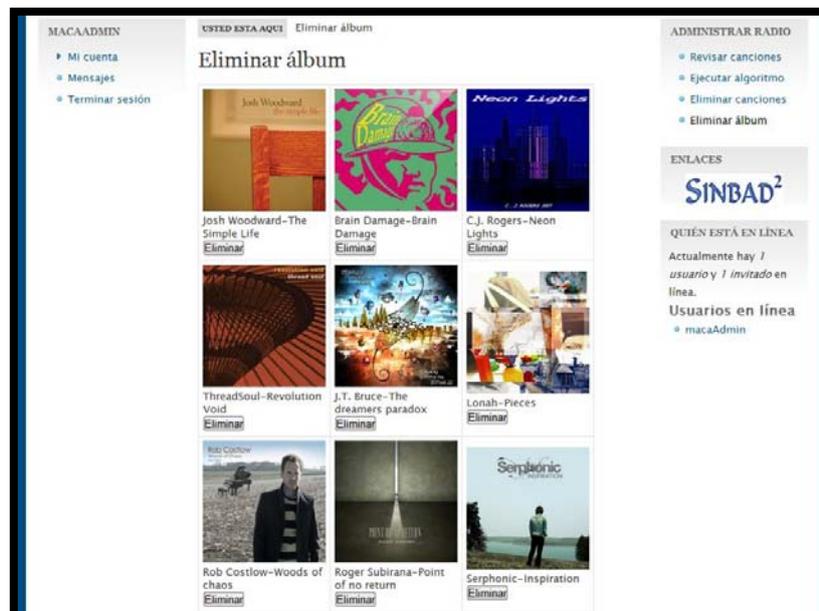


Figura 4.28: Pantalla Eliminar álbumes.

4.4.2.4. Caminos de navegación

Hasta ahora hemos explicado las diferentes pantallas dependiendo el tipo de usuario del sistema, pero no conocemos nada de la interacción entre las pantallas.

Para estudiar los caminos de navegación se empleará una herramienta llamada **storyboard**, que consiste en mostrar, a modo de secuencia, las diferentes pantallas por las que se va pasando al realizar el usuario una determinada acción sobre la aplicación.

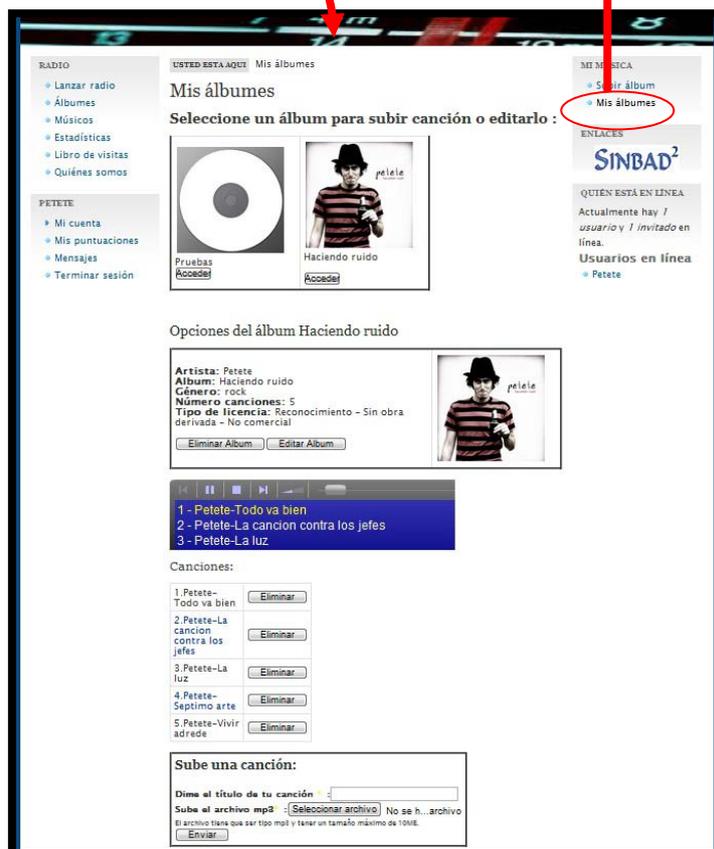
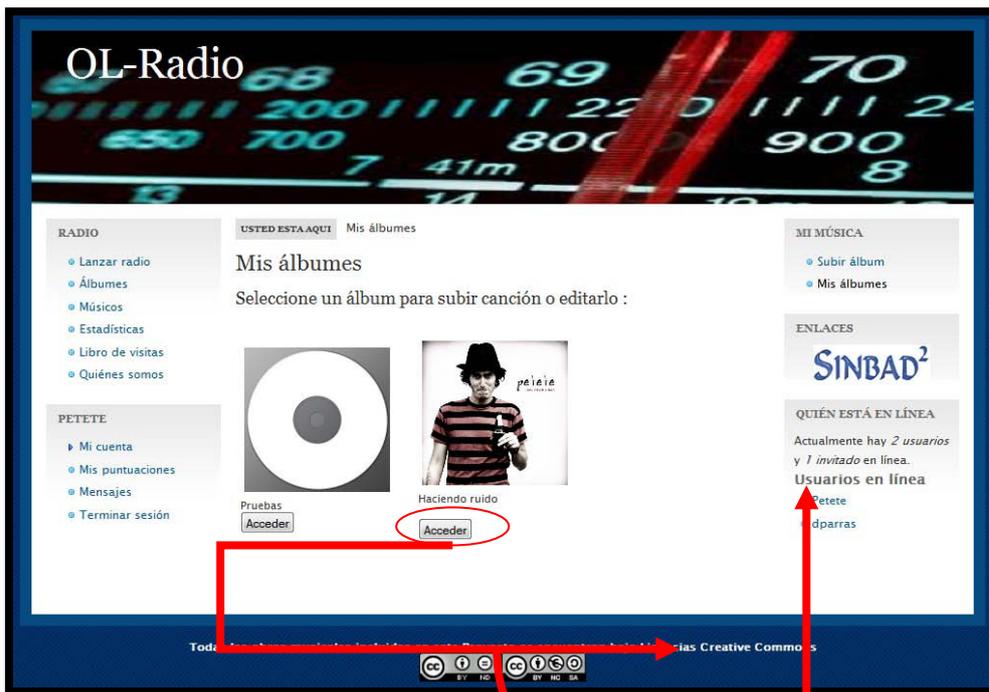
El procedimiento es el siguiente: se sitúan capturas de las pantallas de la interfaz unidas mediante flechas para indicar el camino que sigue la interacción. La posición de origen de las flechas debe ayudar a entender cuál es el elemento que ha desencadenado el paso de una pantalla a otra. Los storyboards también están muy ligados a los escenarios anteriormente vistos.

El storyboard sirve de prototipo para ser evaluado por el usuario y poder introducir correcciones en fases tempranas, ya que cuanto más tiempo se tarde en validar una interfaz, más coste de tiempo y trabajo acarreará.

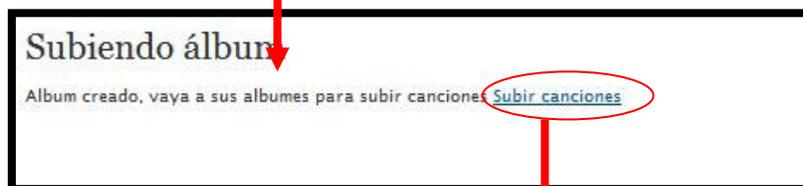
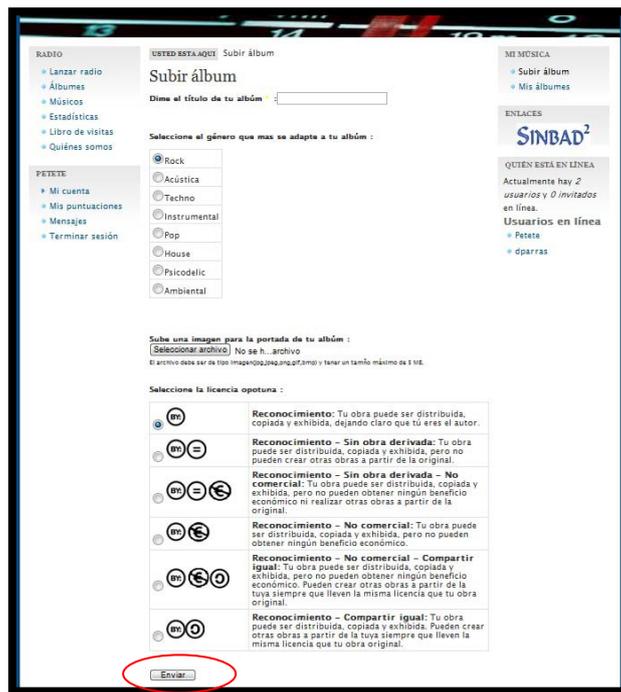
A continuación, mostramos los storyboards para las acciones nuevas más importantes que se pueden llevar a cabo en nuestro sistema.

- Storyboard Mis álbumes.
- Storyboard Subir álbum.
- Storyboard Subir canción.
- Storyboard editar/eliminar álbum
- Storyboard Ver músicos

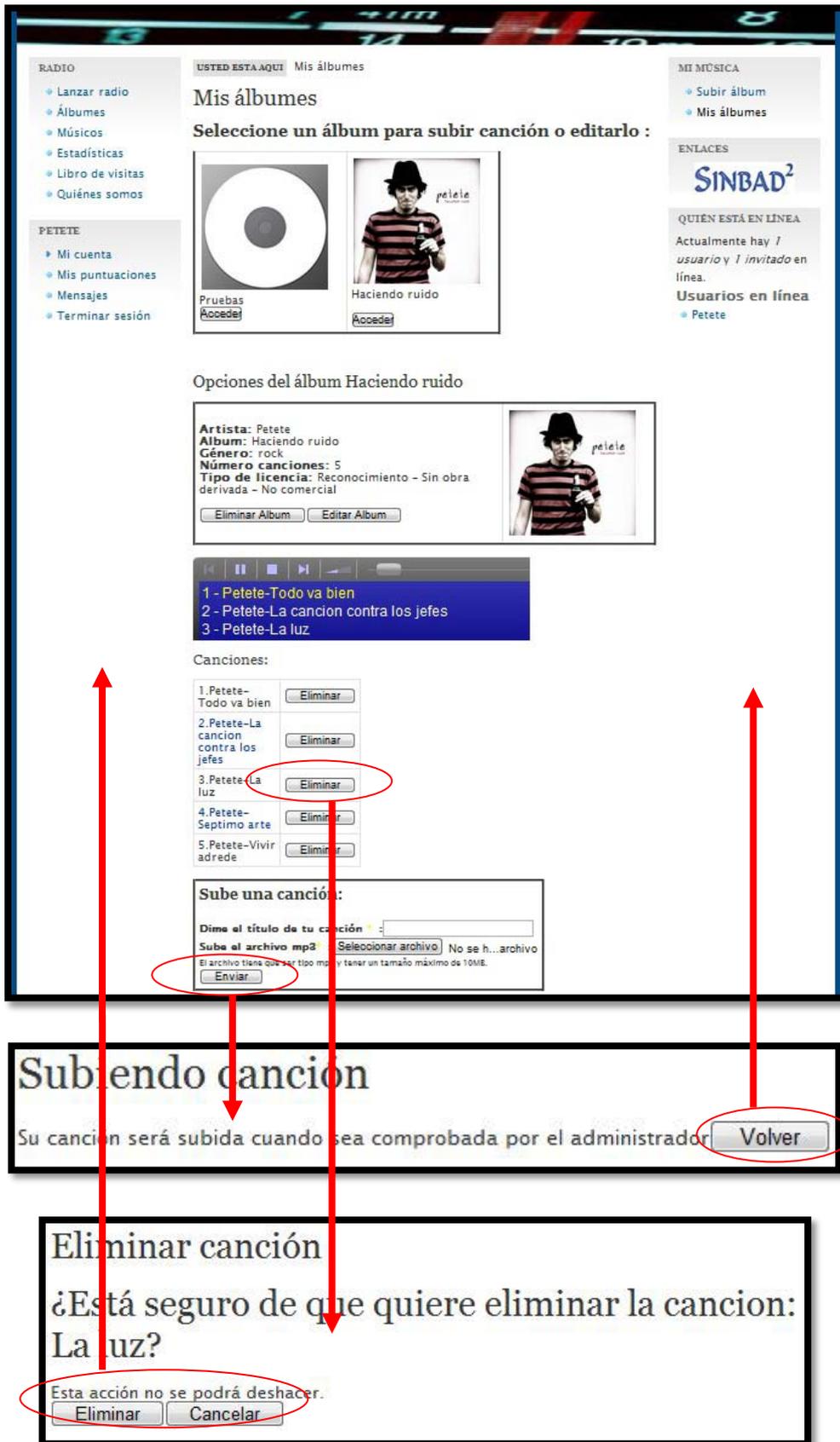
Storyboard Mis álbumes



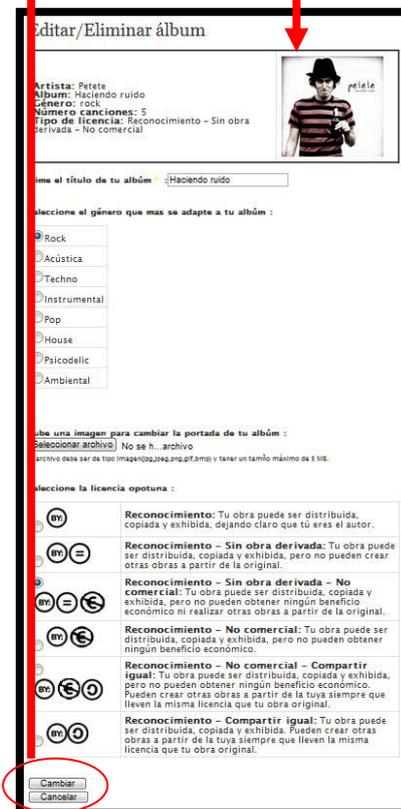
Storyboard Subir álbum



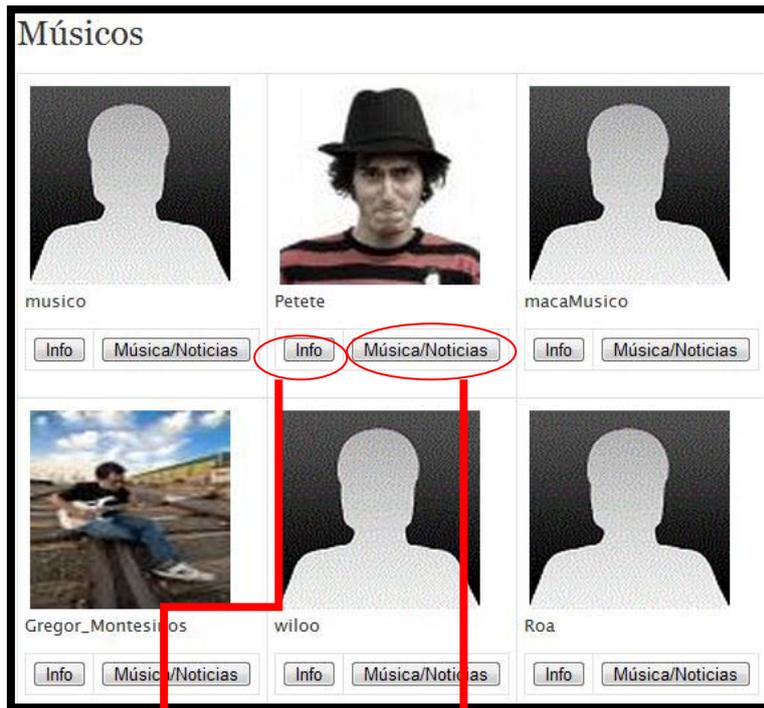
Storyboard Subir canción



Storyboard Editar/Eliminar álbum



Storyboard Ver músicos



4.5. Implementación

La implementación es la actividad final de la Ingeniería del Software, aquella en la que el modelo obtenido en las actividades anteriores se debe transformar en código fuente. Para ello, se debe ser cuidadoso en la elección del lenguaje de programación empleado para la codificación y de la herramienta utilizada para generarla.

4.5.1. Tipo de arquitectura de la aplicación

En el caso de nuestra aplicación, desarrollaremos un Sistema de Recomendación con una arquitectura cliente/servidor y una interfaz Web de comunicación con el usuario. El funcionamiento de este tipo de arquitectura es sencillo: la aplicación se encuentra en un servidor central, al que los usuarios acceden por medio un software cliente, en nuestro caso un navegador Web. Una vez que ha accedido a la aplicación, el usuario realiza peticiones que son atendidas por el servidor, generando así una respuesta adecuada y comprensible para el cliente.

Una arquitectura cliente/servidor (véase Figura 4.29) elimina del usuario final de la aplicación la tarea de tener que instalar dicha aplicación en su máquina, consiguiendo además que cada usuario acceda únicamente a la información que le corresponde. Además, dado su diseño modular, esta arquitectura es fácilmente escalable y ampliable tanto en nuevos clientes como en servidores añadidos.

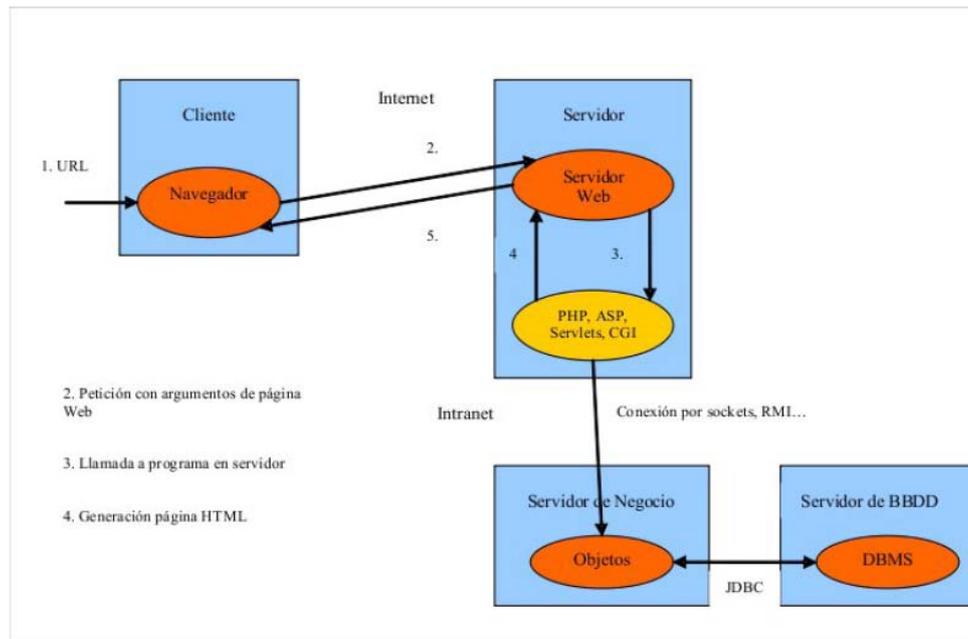


Figura 4.29: Arquitectura Cliente/Servidor

4.5.2. Lenguajes de programación utilizados

Parte Cliente

Ante la arquitectura y el funcionamiento previsto de nuestra aplicación, resulta evidente que el uso de HTML simple y plano no es adecuado, sino que precisa de otros lenguajes capaces de generar contenido dinámico desde el servidor, de manera transparente al usuario final. Nosotros hemos optado por utilizar el lenguaje PHP [12].

PHP, acrónimo recursivo de Hypertext Preprocessor, es un lenguaje de programación interpretado, que se ejecuta del lado del servidor y genera contenido dinámico a petición del cliente.

Se trata de un lenguaje con una serie de importantes ventajas frente a otros lenguajes que realizan funciones parecidas, como son las siguientes:

- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL (de gran utilidad en la implementación de este proyecto).

- Lectura y manipulación de datos desde diversas fuentes, incluyendo datos que pueden introducir los usuarios desde formularios HTML.
- Capacidad de expandir su potenciales gracias al enorme abanico de módulos disponibles (llamados extensiones o ext's).
- Posee una amplia documentación en su página oficial, entre la cual cabe destacar la explicación de todas las funciones del sistema, mediante ejemplos y archivos de ayuda.
- No precisa de la declaración de variables.
- Es libre, por lo que se presenta como una alternativa fácilmente accesible para todos.
- Permite las técnicas de programación Orientada a Objetos.
- Permite la creación de formularios para la Web.
- Amplia biblioteca nativa de funciones incluida. No requiere definición de tipos de variables ni un manejo avanzado a bajo nivel.

Con PHP es suficiente para satisfacer las funcionalidades que debe presentar la aplicación a sus usuarios. Sin embargo, para realizar una adecuada implementación de la interfaz Web se hace necesario el uso de otro lenguaje: CSS.

CSS, acrónimo de Cascade Style Sheets, es un lenguaje formal que ayuda a separar la estructura interna de un documento de su presentación externa. Las etiquetas de estilo CSS pueden presentarse tanto dentro de un documento HTML (encerradas dentro de las etiquetas `<style type="text/css"></style>` en la cabecera) como en un documento aparte (con extensión .css) al que el documento HTML se encarga de llamar cuando es necesario. De esta última manera no sólo se consigue separar la estructura de la presentación sino que también se consigue la centralización del estilo, ya que una sola hoja de estilos CSS puede ser invocada por diferentes páginas de la aplicación Web, lo que ayuda de manera muy importante al mantenimiento de la coherencia y consistencia del diseño de la aplicación.

En nuestra aplicación, el uso de hojas de estilo CSS es algo ineludible dado que así conseguimos que las sentencias PHP del servidor generen, simplemente, el código XHTML necesario para responder a la petición del cliente, sin entrar en temas de diseño o visualización de esta respuesta, de lo que se encargará el estilo CSS predefinido.

Parte Servidor

La implementación de esta parte se ha realizado usando Java y sentencias sql (MySQL) mediante el empleo de la API conocida como JDBC.

Java es un lenguaje de programación orientado a objetos, desarrollado por Sun Microsystems a principios de los 90. El lenguaje en sí toma buena parte de su sintaxis de los lenguajes C y C++, pero tiene un modelo de objetos más sencillo y elimina además herramientas de bajo nivel que pueden inducir a errores, como la manipulación directa de punteros.

Las aplicaciones Java están normalmente compiladas en un bytecode⁴, aunque la compilación en código máquina nativo es también posible. En tiempo de ejecución, el bytecode suele ser interpretado o compilado nuevamente, a código nativo para su ejecución, aunque también es posible la ejecución directa del bytecode vía hardware.

4.5.3. Herramientas de desarrollo

Una vez descritos los lenguajes de programación empleados para este proyecto, pasamos a especificar las herramientas empleadas para el desarrollo de la aplicación a través de dichos lenguajes.

Para generar código Java se requiere de una herramienta o entorno de desarrollo. En nuestro caso, hemos optado por utilizar el entorno de desarrollo

⁴ Un tipo de código intermedio utilizado por Java, de mayor abstracción que el código máquina, que suele ser tratado como un fichero binario ejecutable.

integrado (IDE) **NetBeans 6**. Se trata de una plataforma para el desarrollo de aplicaciones de escritorio usando Java.

La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes software llamados módulos.

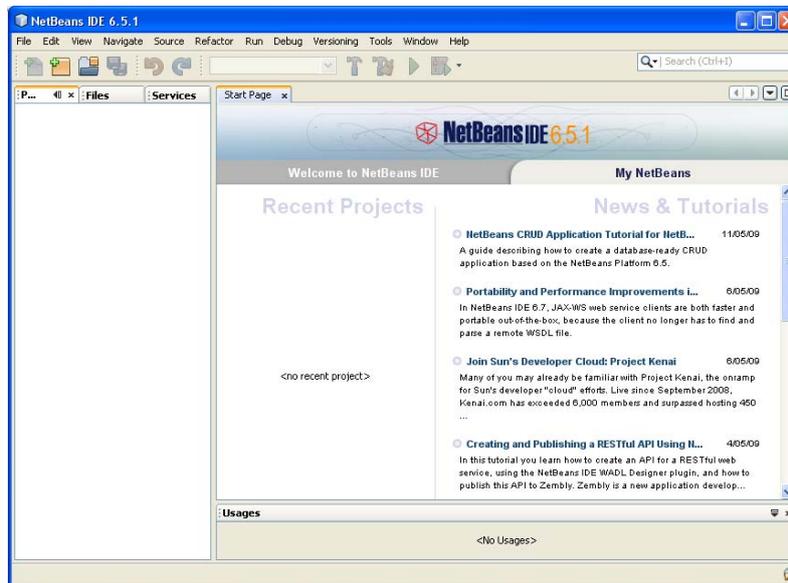


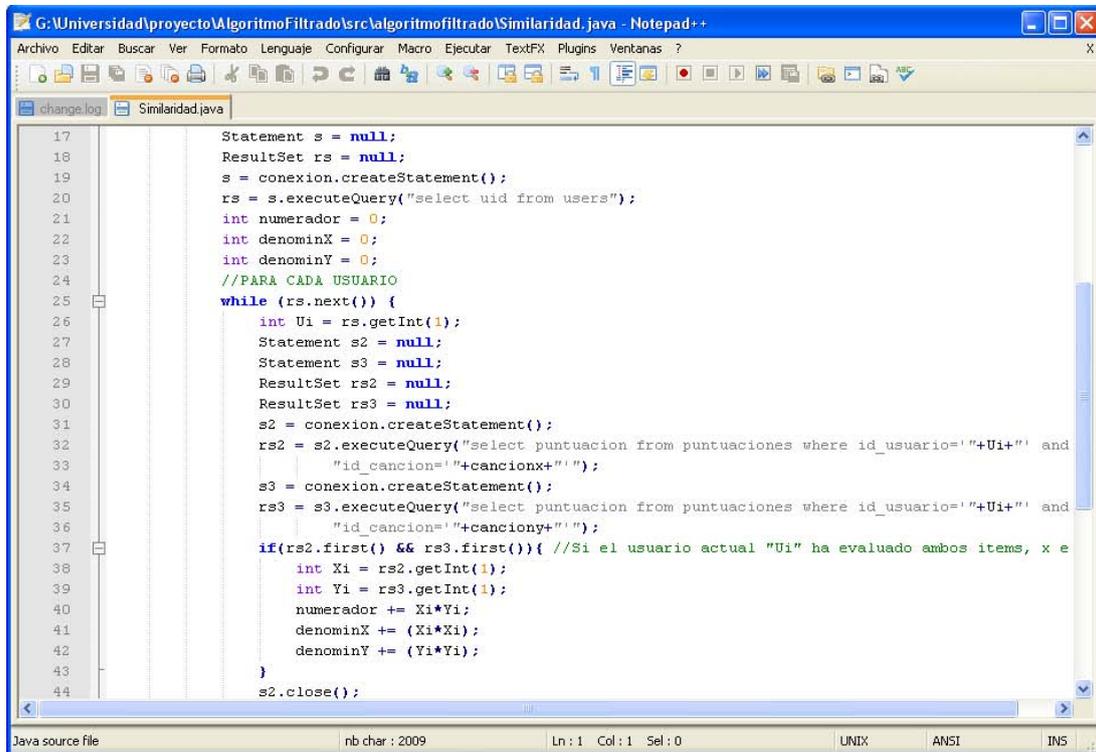
Figura 4.30: Panel de bienvenida de NetBeans 6

NetBeans es un proyecto de código abierto de gran éxito y una enorme cantidad de usuarios en la actualidad, una comunidad en continuo crecimiento y en torno a 100 socios en todo el mundo.

Para la generación de código CSS y xHTML no se requiere una herramienta o entorno de desarrollo en particular. Con un simple editor de textos se pueden escribir las sentencias y etiquetas y guardar el resultado, especificando la extensión correspondiente. El desarrollo de aplicaciones Web mediante estos editores puede ser bastante delicado a veces, ya que son demasiado genéricos y no son capaces de reconocer la sintaxis. Una posible solución es el empleo de editores de textos avanzados como Notepad++.

Notepad++ (Figura 4.31) es un editor de código fuente gratuito, que soporta varios lenguajes de programación y se ejecuta bajo sistemas operativos Windows. Algunas de las características más destacables de Notepad++ son:

- Sintaxis coloreada y envoltura de sintaxis.
- Autocompletado.
- Soporte para buscar/reemplazar expresiones regulares.
- Resaltado de paréntesis y sangrías.
- Grabación y reproducción de macros.
- Multi-Vista.
- Multi-Documento.



```

17 Statement s = null;
18 ResultSet rs = null;
19 s = conexion.createStatement();
20 rs = s.executeQuery("select uid from users");
21 int numerador = 0;
22 int denominX = 0;
23 int denominY = 0;
24 //PARA CADA USUARIO
25 while (rs.next()) {
26     int Ui = rs.getInt(1);
27     Statement s2 = null;
28     Statement s3 = null;
29     ResultSet rs2 = null;
30     ResultSet rs3 = null;
31     s2 = conexion.createStatement();
32     rs2 = s2.executeQuery("select puntuacion from puntuaciones where id_usuario='"+Ui+"' and
33         "id_cancion='"+cancionx+"'");
34     s3 = conexion.createStatement();
35     rs3 = s3.executeQuery("select puntuacion from puntuaciones where id_usuario='"+Ui+"' and
36         "id_cancion='"+canciony+"'");
37     if(rs2.first() && rs3.first()){ //Si el usuario actual "Ui" ha evaluado ambos items, x e
38         int Xi = rs2.getInt(1);
39         int Yi = rs3.getInt(1);
40         numerador += Xi*Yi;
41         denominX += (Xi*Xi);
42         denominY += (Yi*Yi);
43     }
44     s2.close();

```

Figura 4.31: Ejemplo de funcionamiento de NotePad++

Web Developer (Figura 4.32) es un complemento para Firefox que añade una barra de herramientas con menú y centenares de opciones muy útiles orientadas a la programación web. **Web Developer** nos permite obtener información sobre todos los aspectos de la página web en la que estamos, sus imágenes, formularios, CSS y cookies.

Hemos usado esta barra de herramientas porque nos permite cambiar el CSS de nuestra página de una manera interactiva por lo que se hace de una manera muy sencilla las pruebas en el diseño.

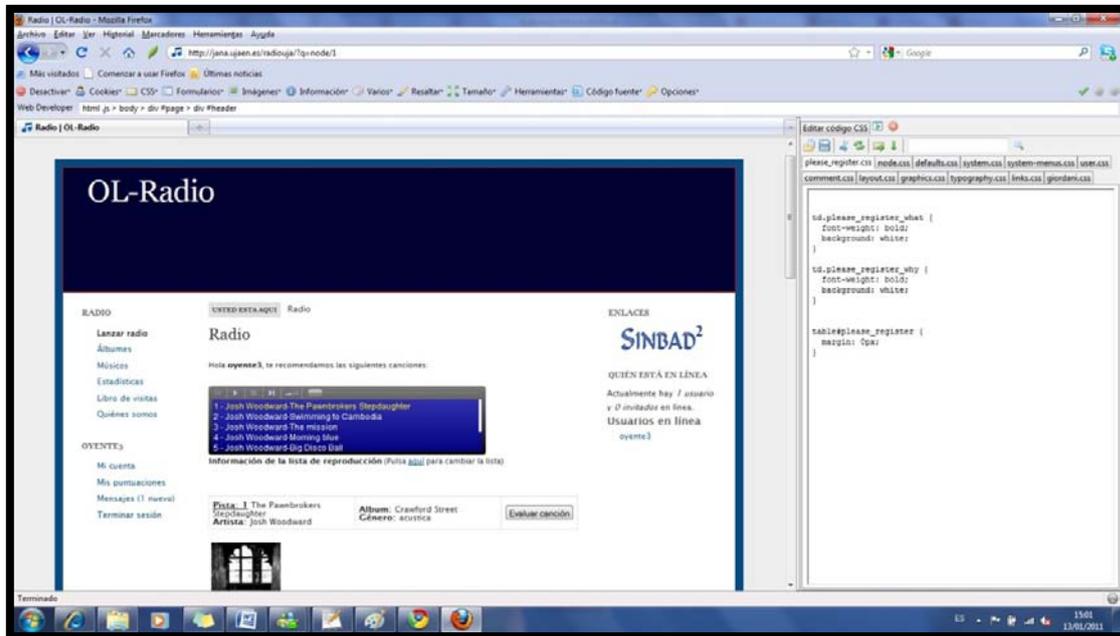


Figura 4.32: Ejemplo modificación de código CSS interactivamente con Web Developer

Para llevar a cabo la funcionalidad PHP y el desarrollo en general de la interfaz de la aplicación se han utilizado dos herramientas perfectamente compatibles:

- Para la construcción del sitio Web se ha utilizado Drupal (Figura 4.33).
- Para el manejo de la base de datos MySQL se ha usado la herramienta phpMyAdmin (Figura 4.34).

Drupal es un Sistema de Gestión de Contenidos (CMS) ampliamente reconocido, con estructura modular y altamente configurable, que permite la construcción de sitios Web en línea potentes.

Es un programa de código abierto, con licencia GNU/GPL, escrito en PHP, desarrollado y mantenido por una amplia comunidad de usuarios. Destaca por la calidad de su código y de las páginas generadas, el respeto por los estándares Web y una especial atención hacia la usabilidad y consistencia de todo el sistema.

El diseño de Drupal es particularmente apto para la construcción y gestión de grandes comunidades en Internet. No obstante, posee una gran flexibilidad y adaptabilidad, lo que unido a la enorme cantidad de módulos adicionales disponibles, lo hace idóneo para diferentes tipos de sitios Web. Hemos añadido cantidad de módulos para aumentar la funcionalidad. Una breve descripción de cada uno de estos módulos adicionales:

- CCK: Permite crear tipos de contenido y formularios de forma muy flexible y altamente configurable.
- Profile: Permite añadir y modificar campos para el perfil de los usuarios.
- Auto Assign Role: Permite a los usuarios elegir el tipo de usuario que quieren ser al realizar su registro.
- Privatemsg: Opciones de envío y recepción de mensajes privados entre usuarios.
- Profile_role: Relacionar categorías del campo de perfil a funciones según los roles de los usuarios.
- Profile permission: Restringir la edición o la visualización de los campos del perfil.
- Node privacy by role: Restringe la visualización/edición de nodos según el rol del usuario.
- Activity Stream: Permite a los usuarios obtener los estados de redes sociales, este modulo estará acompañado de Activity Stream Facebook y Activity Stream Youtube.
- Tweet: Módulo que permite redirigir partes del sitio a la red social Twitter.
- One page profile: Nos da la opción de diseñar los perfiles de usuarios.
- Potx: Para realizar traducciones de idioma en los módulos.
- PHP Filtrer: Permite la evaluación de fragmentos de código PHP.



Figura 4.33: Panel de administración de Drupal

Miles de usuarios de todo el mundo participan activamente en drupal.org, el sitio oficial de desarrollo y coordinación de Drupal.

PhpMyAdmin es una herramienta escrita en PHP con la intención de manejar la administración de MySQL a través de páginas Web, utilizando Internet. Actualmente permite la creación y eliminación de bases de datos, creación, eliminación y modificación de tablas, borrar, editar y añadir campos, ejecutar sentencias SQL, administrar claves en campos, administrar privilegios, y exportar e importar datos en varios formatos. Está disponible en 50 idiomas, y bajo licencia GPL.



Figura 4.34: Pantalla principal de phpMyAdmin

CAPÍTULO 5

Conclusiones

5.1. Conclusiones

En los últimos tiempos, la radio por Internet está siendo uno de los mayores atractivos de ocio y tiempo libre para los internautas en general, debido a su facilidad de uso y su alto grado de accesibilidad.

El proyecto actual nació de la idea de aumentar la funcionalidad del proyecto Radio Online basada en un motor de filtrado colaborativo [28] desarrollado por Iván Palomares en el 2009. Por tanto, en la primera fase del presente proyecto se ha realizado un exhaustivo análisis al anterior y así pudimos detectar algunas carencias o deficiencias que se podían claramente mejorar como describimos en el capítulo primero, la mejora de estas carencias y deficiencias se convierte en los objetivos del presente proyecto. En un principio, nos centramos en mejorar estos problemas y empezaron a surgir multitud de nuevas ideas, como por ejemplo que sean los usuarios los que puedan subir su propia música, de este modo se incrementa por parte de los propios usuarios la base de datos de canciones. Este hecho implica la creación de la figura de un usuario administrador que controle todas las canciones subidas. Además este usuario podría encargarse de otras acciones como la ejecución del algoritmo de filtrado colaborativo.

Por otra parte, debido al incremento de la utilización de las redes sociales en los últimos tiempos nos decidimos a unirlas al proyecto. Desde el principio no pensamos en unirlas de una manera competitiva, ya que las redes sociales que se usan están muy asentadas, sino de una manera cooperativa con ellas. Así conseguíamos, entre otras cosas, que los músicos tengan un aliciente más para subir su música al sitio y darse a conocer mediante OL-Radio conectando su perfil en la radio con sus conexiones en sus redes sociales. Por ejemplo, si un músico sube un nuevo videoclip en Youtube o anuncia un nuevo concierto en Twitter esta información será publicada también en su perfil de OL-Radio. Realizando esta idea, conseguiríamos juntar aspectos que no están unidos en ningún sitio, como son conseguir todos los estados de tus **redes sociales** en el mismo sitio, **subida de tu propia música** y poder disfrutar de la **radio colaborativa**.

Por último una de las prioridades era cambiar la interfaz de OL-Radio y convertirla en un portal moderno y parecido a los portales que tienen tanto éxito en la red hoy en día.

Para el desarrollo de este proyecto se han seguido las etapas de Ingeniería del Software. En primer lugar, se han determinado las propiedades que el sistema debía satisfacer, así como las restricciones de este. Seguidamente se creó un modelo del sistema correcto ampliando el anterior y adaptando o mejorando algunos aspectos del pasado. Finalmente se ha codificado este modelo y se ha instalado en el servidor.

Una de las ventajas con las que he contado, es que gracias al trabajo que realizó el anterior autor del proyecto, la forma en que realizó éste y la multitud de buenos comentarios y documentación que ha dejado, me ha resultado más fácil la mejora o creación de la nueva funcionalidad, además he contado con la disposición del anterior autor en ayudarme o aclararme cualquier cosa que no estuviera clara. Como autor de este proyecto, tengo que decir que ha sido un honor para mí continuar con esta buena idea y colaborar en su ampliación. Por supuesto este proyecto me ha aportado muchísimos conocimientos nuevos además de afianzar anteriores conocimientos en campos como en diseño web, CMS , administración de bases de datos y la propia Ingeniería del Software. Este proyecto ha resultado para mí una experiencia positiva y enriquecedora con miras a la futura incorporación al mundo laboral.

Para concluir, espero que este proyecto cumpla con su objetivo y sirva para que los usuarios disfruten de música con licencias libres y sobre todo los artistas o grupos sin acceso a una discográfica puedan compartir su música con los usuarios y de este modo poder promocionarse de una forma totalmente gratuita. Con vistas a futuras ampliaciones del presente proyecto, dependerá del uso que esté recibiendo OL-Radio. Si este uso es abundante, estas mejoras podrían ir en la línea de seguir adaptando la radio online personalizada a las novedades que vayan surgiendo en la red, mejoras para la gestión de la nueva música o creación de perfiles aun más personalizados.

ANEXO A

Manual de instalación del Servidor

Este manual recoge la instalación del servidor necesario para montar nuestra aplicación. En él, se describirá la instalación de elementos necesarios como el servidor Apache 2, php, mysql, la interfaz phpMyAdmin, la máquina virtual Java y NetBeans.

Consideraciones previas

Es posible, y de hecho normalmente se realiza así, la instalación de cada uno de los citados elementos por separado. No obstante, hemos optado por la instalación de **AppServ**, un paquete software de código abierto que integra todas las aplicaciones, a excepción del entorno de desarrollo y la máquina virtual de Java.

Hemos supuesto para este manual que la unidad principal de disco duro es C:, y que la unidad principal de CD/DVD es D:. Toda la instalación y configuración descrita en este manual se refiere a un servidor montado sobre un Sistema Operativo Windows.

Todo el material necesario para instalar y poner en funcionamiento el servidor se encuentra disponible en el disco que acompaña a esta memoria.

A.1. Instalación de AppServ

AppServ permite, de una forma fácil y rápida, la instalación de los siguientes elementos:

-Apache

-PHP

-MySql

-PhpMyAdmin.

Se puede descargar en la dirección <http://www.appservnetwork.com>, siendo la versión 2.5.10 la última versión estable en el momento de la realización de este manual,

incluida además en el disco que acompaña a esta memoria. Dicha versión incluye las siguientes herramientas (ver Figura A.1):

- Apache 2.2.8
- PHP 5.2.6
- MySQL 5.0.51b
- PhpMyAdmin 2.10.3

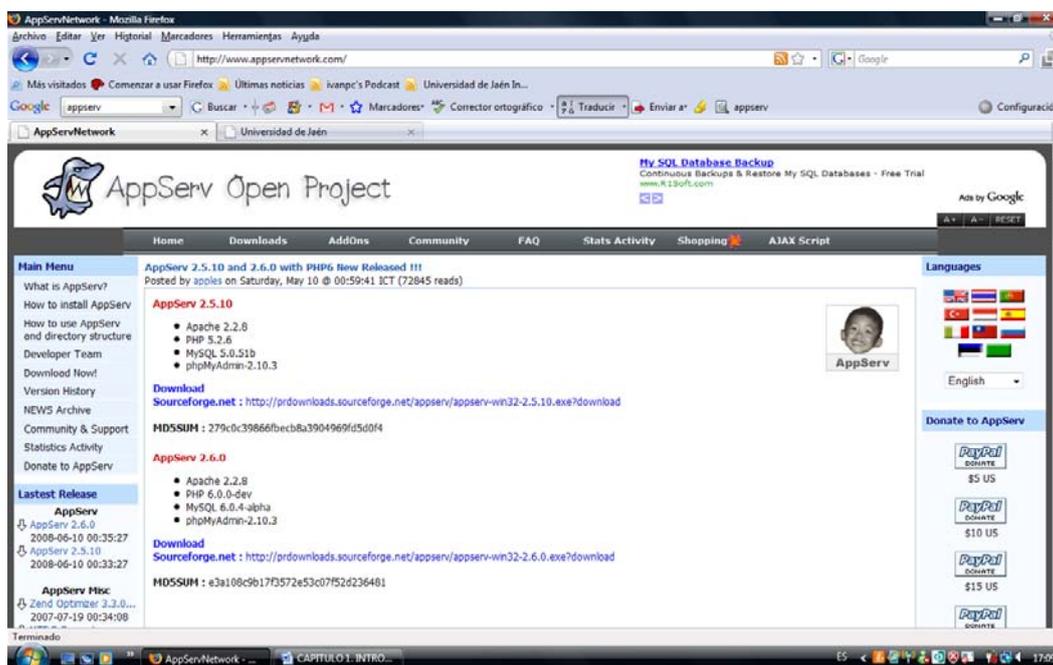


Figura A.1: Portal deAppservNetwork.com

Para descargar AppServ, basta con hacer clic sobre el enlace de descarga y elegir un directorio de destino en nuestro equipo para el instalador. Una vez descargado ejecutaremos el instalador haciendo clic sobre el archivo descargado *appserv-win32-x.x.x.exe* y nos aparecerá la siguiente ventana (Figura A.2).



Figura A.2: Instalador de AppServ

Al pulsar el botón 'Next', aparecerá el acuerdo de licencia que, tras leer con detenimiento, deberá ser aceptado, mediante la opción 'I Agree' (Figura A.3), para continuar.

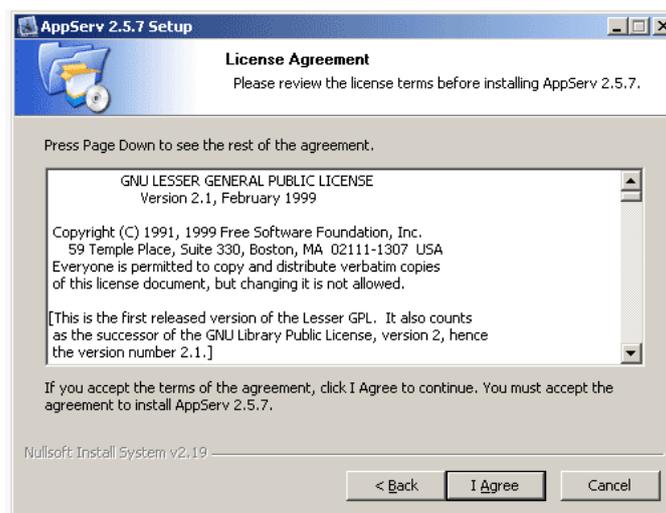


Figura A.3: Acuerdo de licencia de AppServ

A continuación el usuario debe elegir un directorio para la instalación de la aplicación, (normalmente aparecerá la ruta por defecto C:\AppServ) y seguidamente pulsar 'Next' (Figura A.4).

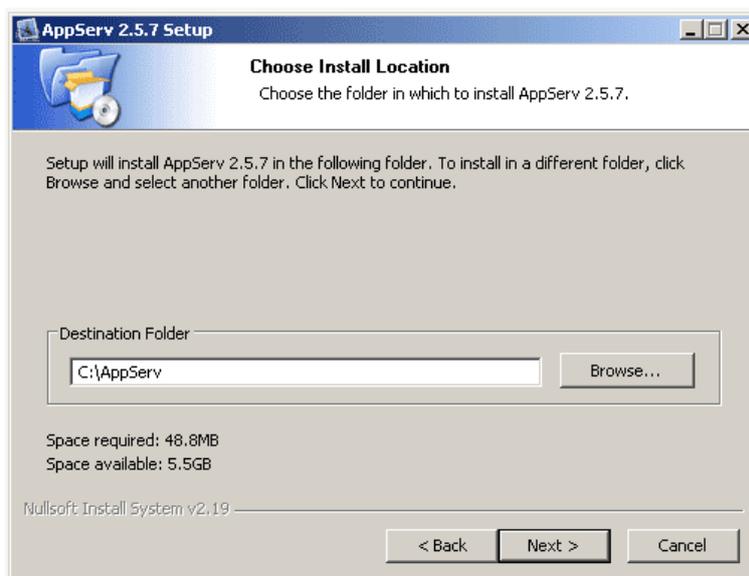


Figura A.4: Selección del directorio de instalación

La siguiente pantalla que aparece es la pantalla de selección de componentes a instalar (Figura A.5). Por defecto aparecen todos los componentes seleccionados para ser instalados. En nuestro caso, utilizaremos esta opción, por lo que solo habrá que pulsar nuevamente sobre 'Next'.



Figura A.5: Pantalla de selección de componentes

El siguiente paso es la configuración del servidor Apache (Figura A.6). AppServ se encarga por sí solo de realizar casi todas las configuraciones de forma automática, por lo que el usuario solo debe preocuparse de introducir los siguientes campos:

- Server Name (Nombre del servidor): Normalmente será 'localhost'.
- Administrator's email address: Una dirección de correo válida del administrador del servidor.
- Apache HTTP Port: Puerto de escucha del servidor, dejar el puerto 80 asignado por defecto.

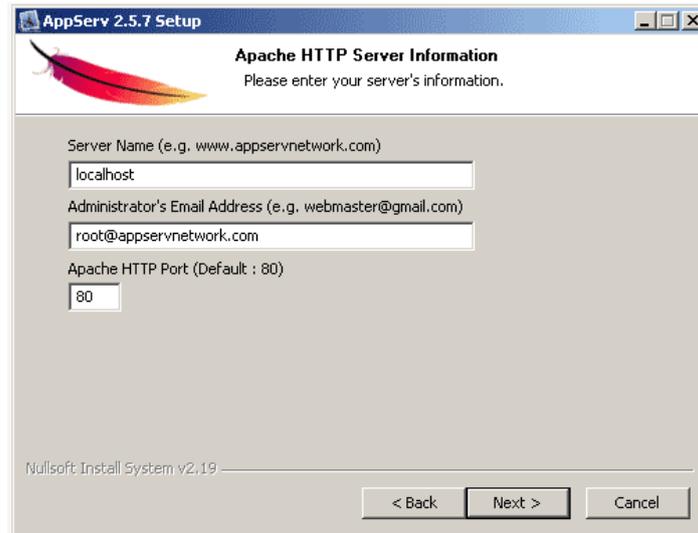


Figura A.6: Configuración del Servidor Apache

Una vez ha quedado definida la configuración de nuestro servidor Apache, el siguiente paso consiste en configurar el componente MySQL Server. Nos aparecerá la ventana mostrada en la Figura A.7:



Figura A.7: Configuración de MySQL Server

En nuestro caso, basta con introducir dos veces la contraseña para nuestra cuenta de usuario administrador 'root', que es el usuario por defecto que AppServ crea para MySQL. La contraseña de la base con la que nuestra aplicación viene preparada para funcionar es '**administrador**'. El resto de opciones se puede dejar con los valores asignados por defecto, por lo que una vez introducida una contraseña pulsaremos sobre 'Next'.

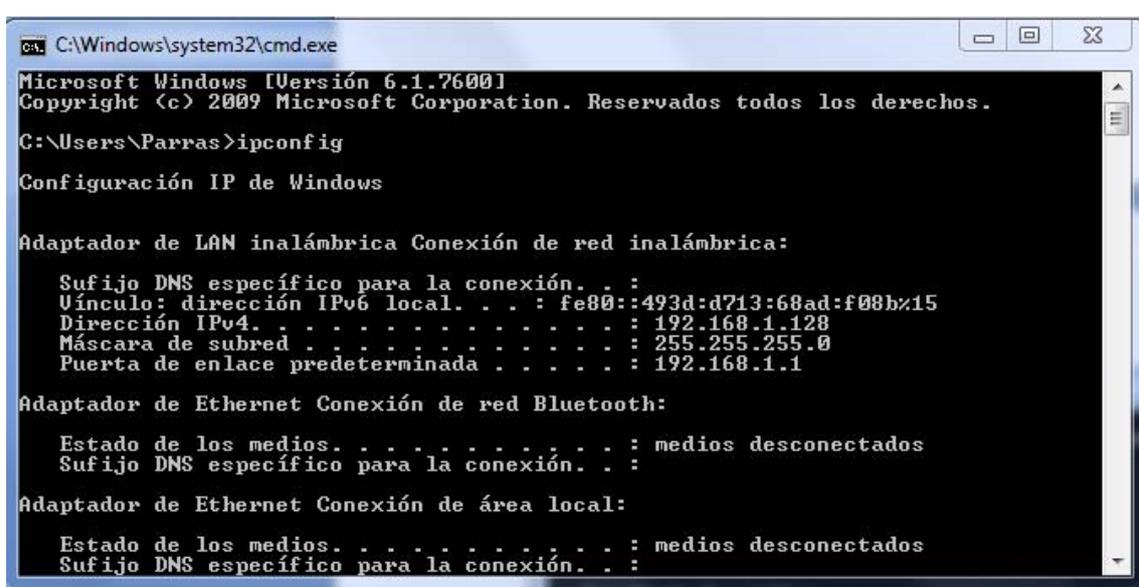
Tras finalizar la instalación (Figura A.8), se nos preguntará si deseamos comenzar la ejecución de Apache y MySql de inmediato, aceptamos y pulsamos sobre el botón 'Finish'.



Figura A.8: Pantalla fin de instalación de AppServ

Una vez instalado AppServ y comenzado la ejecución del servidor Apache y MySQL, es importante saber cómo acceder a nuestro servidor desde el navegador.

Para ello basta con escribir nuestra dirección IP en el navegador. En caso de no saberla, se puede averiguar fácilmente ejecutando la línea de comandos de Windows (cmd), tecleando la orden 'ipconfig'. El campo 'DirecciónIPv4' nos da el número de nuestra dirección IP (Figura A.9).



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Users\Parras>ipconfig

Configuración IP de Windows

Adaptador de LAN inalámbrica Conexión de red inalámbrica:

    Sufijo DNS específico para la conexión. . . :
    Vínculo: dirección IPv6 local. . . . . : fe80::493d:d713:68ad:f08b%15
    Dirección IPv4. . . . . : 192.168.1.128
    Máscara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada . . . . . : 192.168.1.1

Adaptador de Ethernet Conexión de red Bluetooth:

    Estado de los medios. . . . . : medios desconectados
    Sufijo DNS específico para la conexión. . . :

Adaptador de Ethernet Conexión de área local:

    Estado de los medios. . . . . : medios desconectados
    Sufijo DNS específico para la conexión. . . :
```

Figura A.9: Obtención de la dirección IP

Tras introducir la dirección IP en el navegador, si todo ha ido correctamente, debería aparecernos una pantalla similar a la mostrada en la Figura A.10.

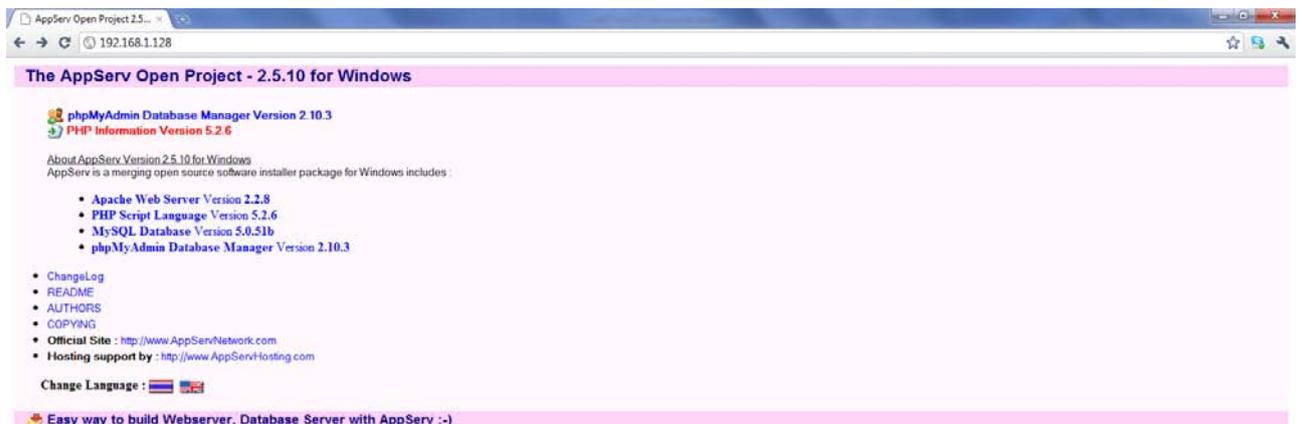


Figura A.10: Pantalla inicial del servidor AppServ

Desde aquí se puede acceder a las principales funcionalidades de los componentes instalados. En particular, nos interesa el acceso a la base de datos, para lo cual hacemos clic sobre el enlace 'phpMyAdmin Database Manager Version X.X.X.'. Se nos pedirá usuario y contraseña para continuar (Figura A.11). La contraseña es la que introducimos durante la instalación de MySQL (administrador), y el usuario es 'root'.

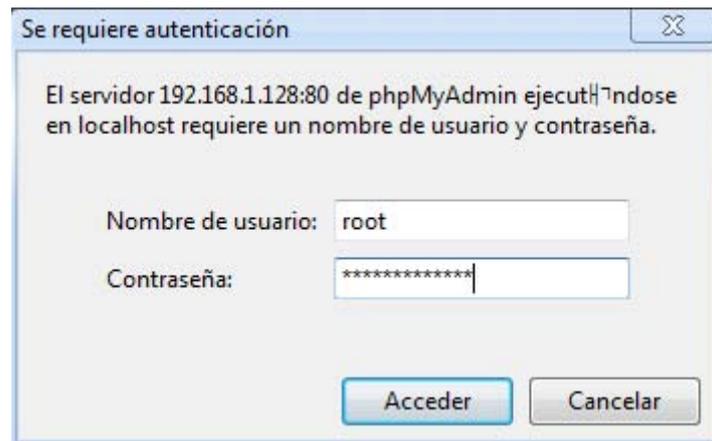


Figura A.11: Control de acceso a phpMyAdmin

Una vez identificado, aparecerá la pantalla principal de la interfaz phpMyAdmin, desde la que se puede administrar la base de datos.



Figura A.12: Pantalla de inicio de phpMyAdmin en el servidor

A.2. Alojamiento del contenido de la aplicación en el servidor

Una vez que hemos terminado con la instalación del servidor, tendremos que alojar nuestra aplicación dentro del mismo.

En la configuración estándar de AppServ el directorio de instalación se encuentra en C:\AppServ, la ubicación que equivale a la dirección <http://localhost> o [http://\[DirecciónIP\]](http://[DirecciónIP]) es C:\AppServ\www (donde reemplazaremos [DirecciónIP] por la dirección de red de nuestro equipo). Por tanto, habremos de copiar la carpeta OL-Radio dentro de C:\AppServ\www.

A.3. Importar la base de datos y activar Drupal

Para importar el archivo con la base de datos, 'pfc-radio.sql', debemos acceder a nuestra interfaz de phpMyAdmin y, en la pantalla principal, seleccionamos la opción 'Crear nueva base de datos'. En la casilla de texto escribiremos el mismo nombre del archivo, 'pfc-radio', y pulsamos el botón Crear (ver Figura A.12). Seguidamente, seleccionamos la opción 'importar' situada en la parte inferior de la pantalla inicial.

Servidor: localhost

Bases de datos SQL Estado actual Variables Juegos de caracteres Motores Privilegios Procesos Exportar Importar

Importar

Archivo a importar

Localización del archivo de texto: C:\Users\lván\Desktop\pfc-radio.sql (Tamaño máximo: 81,920KB)

Juego de caracteres del archivo: utf8

La compresión escogida para el archivo a importar se detectará automáticamente de: Ninguna, gzip, zip

Importación parcial

Permita la interrupción de la importación en el caso de que el script detecte que se ha acercado a su límite de tiempo. Esto podría ser un buen método para importar archivos grandes; sin embargo, puede dañar las transacciones.

Número de registros (consultas) a saltarse desde el inicio: 0

Formato del archivo importado

SQL

Opciones SQL

Modalidad compatible con SQL: NONE

Continuar

[Abrir nueva ventana de phpMyAdmin](#)

Figura A.13: Importación de la base de datos.

Para cargar el archivo solamente tenemos que buscarlo mediante el botón Examinar y, acto seguido, pulsar el botón Continuar situado en la esquina inferior derecha (Figura A.13).

La instalación de Drupal queda realizada con sólo copiar la carpeta OL-Radio en el directorio del servidor C:\AppSer\www. Esto es así porque hemos incluido en el material todos los archivos de Drupal de forma que el sitio Web quede intacto al migrarlo de un equipo a otro.

Para terminar, teclearemos la siguiente dirección en el navegador, sustituyendo [DirecciónIP] por la dirección IP de nuestro equipo.

[http://\[DirecciónIP\]/OL-Radio](http://[DirecciónIP]/OL-Radio)

Si todo ha ido bien, debería aparecernos la página principal de la aplicación.



Figura A.14: Página principal de la aplicación

A fin de que el usuario pueda disfrutar de inmediato de la aplicación como usuario registrado, la base de datos incluye por defecto un usuario administrador del sitio web, un usuario de tipo oyente, uno de tipo músico y un administrador radio. Las credenciales para cada uno de ellos son:

Identificador administrador: root. **Contraseña administrador:** administrador.

Identificador usuario: oyente. **Contraseña usuario:** oyente.

Identificador músico: musico. **Contraseña músico:** musico.

Identificador administrador radio: admin. **Contraseña administrador radio:** admin.

ANEXO B

Manual de Usuario

En este anexo mostramos, a modo de visita guiada, un manual de la aplicación de utilidad para los distintos usuarios. Se pretende, así, que estos puedan aprender la utilización y poder sacar el máximo partido de la aplicación de la forma más sencilla posible.

No obstante, es conveniente en primer lugar dejar claros algunos aspectos importantes para el usuario:

- La aplicación de radio online OL-Radio es una aplicación Web recomendada especialmente para las últimas versiones existentes de Google Chrome (que se puede descargar en <http://www.google.com/chrome?hl=es>), Mozilla Firefox (que se puede descargar en <http://www.mozilla-europe.org/es/>) y de Internet Explorer. Se recomienda una resolución de 1024x768 píxeles o superior, para evitar posibles fallos de visualización. A pesar de estos posibles fallos en la presentación del sitio Web, la funcionalidad no se verá en absoluto afectada.
- Es importante tener instalado el plugin de reproducción de contenido Flash en el navegador, para poder visualizar la radio. En caso de no disponer de él, la mayoría de versiones tanto de IE como de Firefox muestran una barra emergente en la parte superior de la ventana, bajo las barras de tareas, para brindar al usuario la posibilidad de instalarlo.
- Por último, téngase en cuenta que, pese a que el sitio está hecho para mostrar los paneles de puntuaciones de la radio como ventanas emergentes, se han encontrado versiones de Firefox y Google Chrome en las que en lugar de ventanas emergentes aparecen los paneles de puntuación como nuevas pestañas de navegación.

Para empezar haremos una visita para la funcionalidad de un usuario de tipo **oyente**, seguidamente para la funcionalidad de un usuario de tipo **músico** y para terminar con la funcionalidad para un usuario de tipo **administrador radio**. Para empezar accederemos a la aplicación como usuario anónimo.

B.1. Validación de un usuario

El usuario debe acceder a través de su navegador a la siguiente dirección: <http://jana.ujaen.es/radiouja/>⁵. Al hacerlo, aparecerá la página de inicio de la aplicación, como podemos observar en la Figura B.1.



Figura B.1: Página de inicio de OL-Radio

El usuario deberá dirigirse a la sección 'Inicio de sesión', situada en la parte derecha de la aplicación, introducir su identificador de usuario y contraseña, y pulsar el botón 'Iniciar sesión' (Figura B.2). Llegados a este punto, pueden ocurrir dos cosas. Si el identificador y contraseña son correctos, aparecerá la pantalla principal con privilegios de usuario registrado, que corresponde a la radio personalizada del usuario en este caso de tipo **oyente** (Figura B.3).

⁵ Dirección Web en la que la aplicación ha quedado alojada en el momento de la realización de esta memoria. En caso de haber instalado la aplicación según el Anexo A, la dirección Web será la especificada en dicho anexo.

Inicio de sesión

Usuario: *

Contraseña: *

Iniciar sesión

- ▶ Regístrate
- ▶ Solicitar una nueva contraseña

Figura B.2: Formulario de inicio de sesión

OL-Radio

68 69 70
200 1111 220 1111 24
650 700 800 900
7 41m 8

RADIO

- Lanzar radio
- Álbumes
- Músicas
- Estadísticas
- Libro de visitas
- Quiénes somos

OYENTE3

- ▶ Mi cuenta
- Mis puntuaciones
- Mensajes
- Terminar sesión

USTED ESTA AQUÍ Radio

Radio

Hola **oyente3**, te recomendamos las siguientes canciones:

1 - Josh Woodward-Shower scene
2 - Josh Woodward-The bottom
3 - Maya Filipic-Stories from Emona I
4 - Josh Woodward-The spirit world
5 - Josh Woodward-Revolution Now

Información de la lista de reproducción (Pulsa [aquí](#) para cambiar la lista)

Pista: 1 Shower scene
Artista: Josh Woodward
Album: Crawford Street
Género: acustica

Pista: 2 The bottom
Artista: Josh Woodward
Album: The Simple Life
Género: acustica

Pista: 3 Stories from Emona I
Artista: Maya Filipic
Album: Between Two Worlds
Género: acustica

ENLACES

SINBAD²

QUIÉN ESTÁ EN LÍNEA

Actualmente hay 1 usuario y 1 invitado en línea.

Usuarios en línea

- oyente3

Figura B.3: Pantalla principal para usuario autenticado.

Si, por el contrario, el identificador de usuario o contraseña introducidos son incorrectos, aparecerá un mensaje de error en la pantalla inicial, tal y como muestra la Figura B.4.



Figura B.4: Error de validación en OL-Radio

Si el usuario no dispone de un identificador para validarse en la aplicación, es posible registrarse en tiempo de ejecución, mediante la opción 'Regístrate' situada en la parte inferior del formulario de inicio de sesión.

Al hacer clic sobre dicha opción, aparecerá la siguiente pantalla:

Figura B.5: Registro de usuario.

Para registrarse sólo es necesario introducir un nombre de usuario, una dirección de correo electrónico válida, y seleccionar el tipo de usuario que se desea ser en este caso ponemos usuario de tipo **oyente**.

Si los datos han sido introducidos correctamente, aparecerá la pantalla de inicio con el siguiente mensaje en la parte superior.

Tanto su contraseña como las instrucciones adicionales han sido enviadas a su correo electrónico.

A continuación el nuevo usuario debe acceder a su bandeja de correo electrónico para abrir el mensaje que habrá recibido de la aplicación y consultar su

contraseña, que será la que utilizará para validarse y entrar como usuario autenticado.

Una vez validados en la aplicación, es hora de utilizarla, para ello nos encontramos con dos menús diferenciados, uno será las funciones de radio y otro de funcionalidad específica del usuario (como muestra la Figura B.6).



Figura B.6: Menús de OL-Radio para usuario oyente

B.2. Utilización de las opciones de Menú Radio (usuario registrado)

Vamos a analizar cada una de las funciones de este menú. Podemos decir que este menú estará disponible para cualquier usuario registrado (**oyente** o **músico**) que visite la página.

Lanzar Radio

Como se ha comentado, la radio se encuentra como pantalla principal de la aplicación para los usuarios de tipo **oyente** o usuarios **anónimos**. Además, es accesible desde la opción de menú Radio 'Lanzar radio', que también sirve para actualizar la lista de reproducción en caso de que el usuario ya se encuentre en esta pantalla.

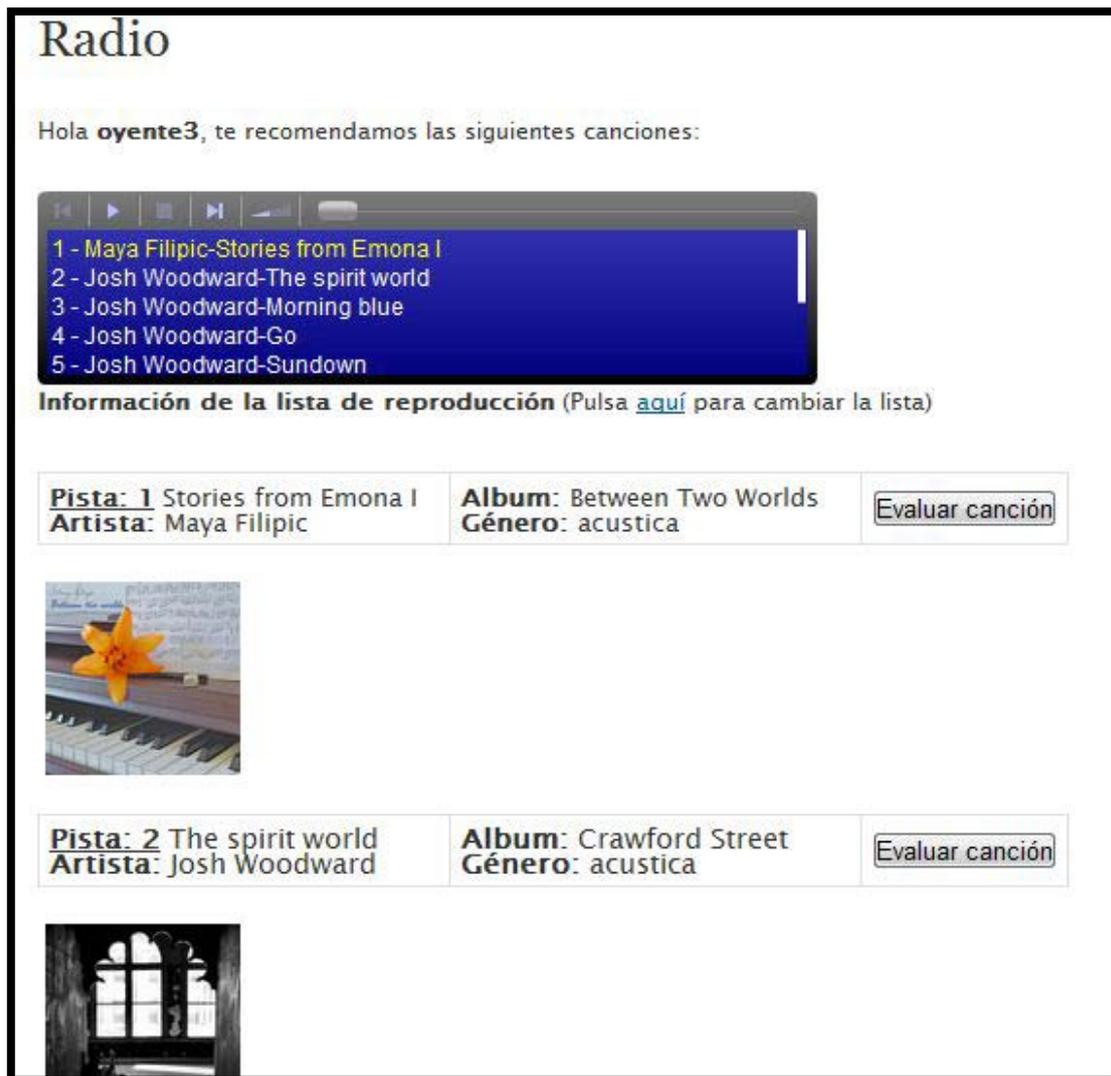


Figura B.7: Detalles de la pantalla de radio.

Esta sección permite al usuario escuchar una lista de canciones que le han sido recomendadas, por ser similares a otras canciones que este ha valorado positivamente con anterioridad.

Escuchar la radio es tan sencillo como utilizar los botones existentes en la lista de reproducción para iniciar, detener o pausar la reproducción, pasar a la canción anterior o siguiente, y ajustar el nivel de volumen, como podemos ver en la siguiente Figura.

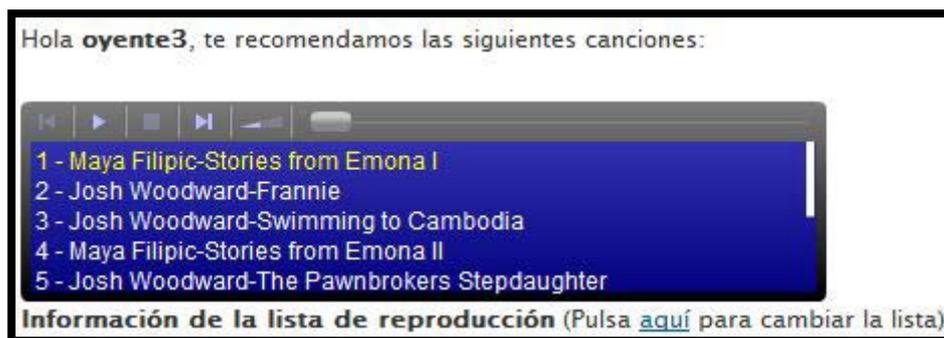


Figura B.8: Lista de reproducción (radio).

El usuario también puede pulsar el enlace situado bajo la radio para cambiar la lista y obtener un nuevo conjunto de canciones recomendadas.

Debajo de la radio aparecen los detalles de la lista de reproducción, donde se puede ver, entre otros datos, el nombre de la canción, género o carátula del álbum al que pertenece.



Figura B.9: Detalles de canción en la radio del usuario.

Si el usuario lo desea, puede evaluar la canción, que consiste en asignarle una puntuación de 1 a 5, según la siguiente escala de valores:

- 1: Mala.
- 2: Regular
- 3: Buena
- 4: Muy Buena
- 5: Excelente.

Al pulsar sobre 'Evaluar canción', aparecerá una ventana emergente⁶ en la que el usuario puede asignar la puntuación deseada a la canción (Figura B.10). Basta con elegir uno de los 5 valores y pulsar sobre el botón 'Puntuar'.

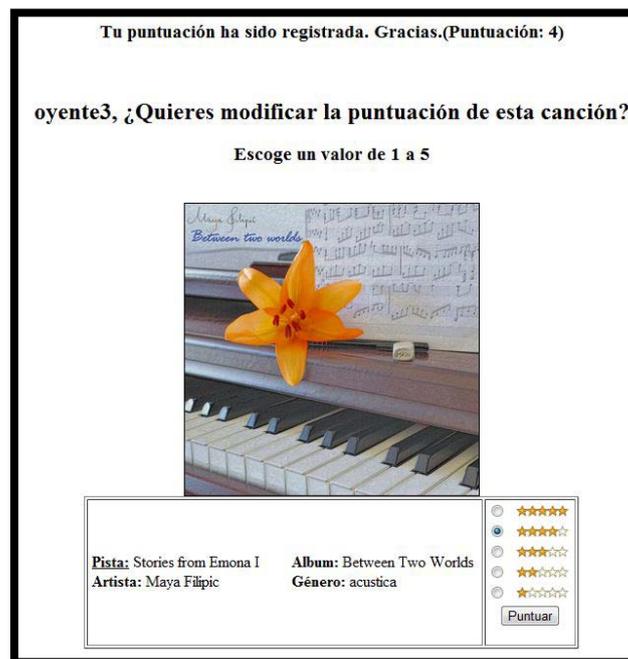
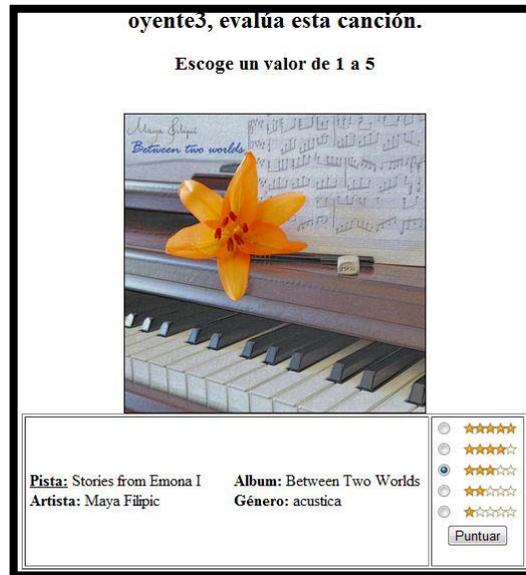


Figura B.10: Pantalla de puntuación de canciones.

Todo este proceso de puntuación se puede hacer cómodamente mientras el usuario sigue escuchando su radio, sin que la reproducción de esta se vea

⁶ O pestaña, en algunas versiones de Mozilla Firefox y Google Chrome

interrumpida. Una vez puntuada la canción, se puede cerrar la ventana de puntuaciones y continuar disfrutando de la aplicación.

El usuario **anónimo** también puede disfrutar de la radio de la aplicación, con la peculiaridad de que no podrá evaluar canciones, y de que la lista de reproducción lanzada estará compuesta por canciones escogidas de forma totalmente aleatoria.

Álbumes

Esta opción del menú permite acceder a la colección de álbumes disponible en la aplicación. Por defecto aparece una vista global de todos los álbumes en orden alfabético de artista (Figura B.11).



Figura B.11: Vista principal de la sección 'Álbumes'

También es posible seleccionar un género musical para mostrar únicamente aquellos álbumes que pertenecen a dicho género (ver Figura B.12), para ello, nos vamos

a 'Vista por género', en la parte superior de la lista de álbumes, seleccionamos un género y pulsamos 'Aceptar'.

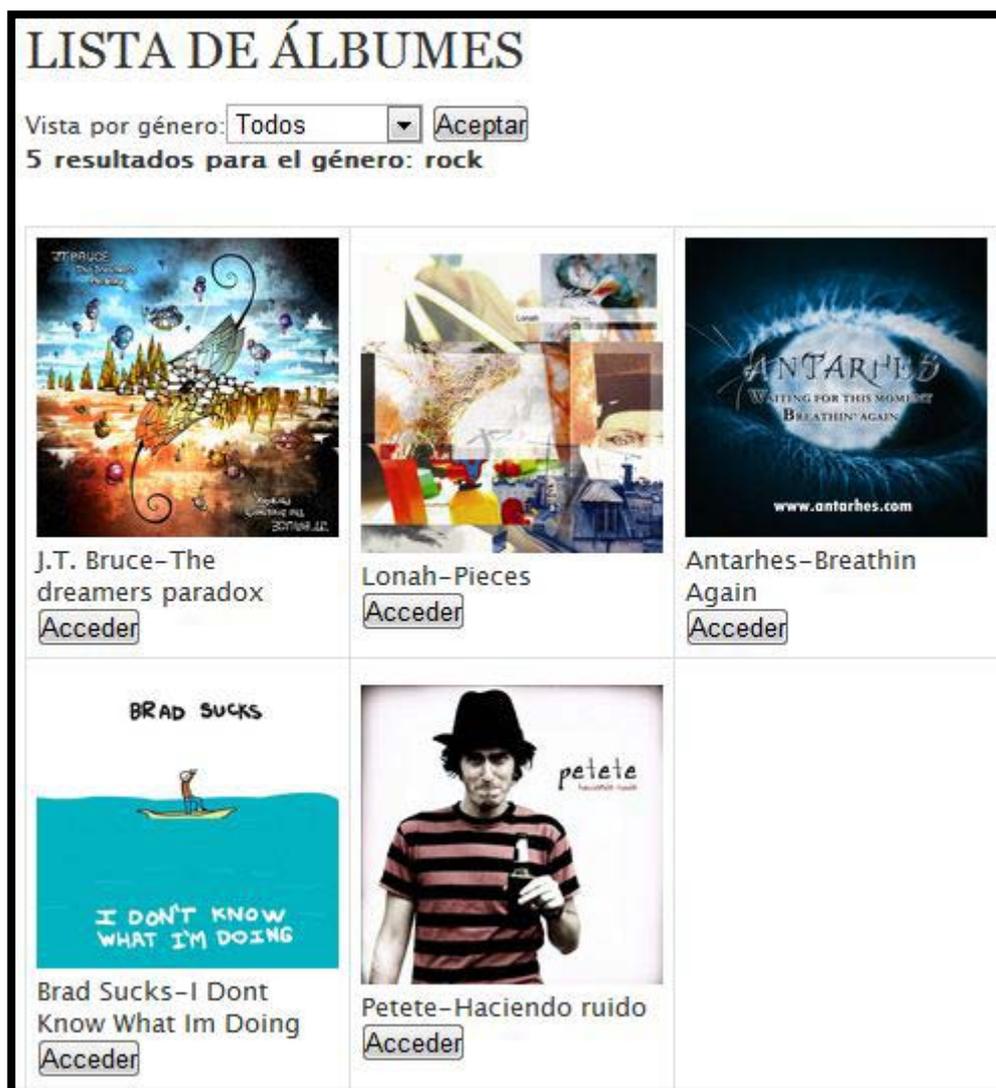


Figura B.12: Vista de los álbumes por género

Para ver la información de un álbum, tan solo hay que hacer clic sobre el botón 'Acceder'. Nos aparecerá así la pantalla de detalles del álbum (Figura B.13), con datos del mismo, y una reproductor con cada una de las canciones del álbum. Además, también se ofrece desde aquí la posibilidad de evaluar las canciones que forman parte del álbum. Para evaluar una canción elegimos en la parte inferior la canción que queramos y pulsamos 'Puntuar'.



Figura B.13: Página de información de álbum

El usuario **anónimo** también acceder al álbum que desee, con la limitación de no poder evaluar canciones.

Músicos

Aquí podemos ver la lista de los músicos registrados en OL-RADIO (Figura B.14) y unas opciones para ver sus noticias, música o su perfil que analizaremos a continuación.



Figura B.14: Página con la lista de músicos

Si pulsamos sobre 'Info ' de cualquier músico podemos acceder a su perfil (Figura B.15) donde podremos ver información detallada del músico (siempre que el músico lo haya escrito anteriormente) como su género, su sitio web, una biografía y algunas de sus conexiones con redes sociales.

Petete

Información general

- Género**
Acústica / Fusión / Rock
- Sitio Web**
<http://www.petete-laband.com/>
- Bio**
Antonio Hernando "Petete" nace en Jaén. Desde niño muestra un gran interés por la música, escuchando vinilos paternos y más tarde, enganchándose tardes enteras a los famosos programas de radio. A los 14 empieza a componer sus primeras canciones a guitarra y armónica, compaginando más adelante este "hobby" con su carrera universitaria. Tras varios años de conciertos por distintas ciudades (Cádiz, Jaén, Sevilla, Valladolid, Madrid, etc.), colaboraciones, canciones, viajes... y una vez terminada la carrera, Petete decide cambiar de aires y trasladarse a Oxford, cuna de grupos de la talla de Radiohead o Supergrass y con una vida y cultura musical que dista mucho de la española. Es en Oxford donde empieza a tocar en Open Mics e incluso en la calle; en una céntrica plaza de la famosa ciudad universitaria. A conocer a músicos de muy diferentes estilos y empapándose absolutamente de todo lo que ocurre a su alrededor. Al poco tiempo de aterrizar en Inglaterra conoce a una persona con la que comparte su gran pasión por la música y que le ayudará infinito en su futuro, además de pasar a ser el guitarrista oficial de La Band, rubén deGracia. Uno componiendo y otro arreglando le dan forma a los temas que, tras dos meses de duro trabajo, dan forma a "Haciendo Ruido", su album debut. Aquí se demuestra y quedan plasmadas las ganas de dejar constancia del nuevo sonido que Petete llevaba buscando durante años y que finalmente se hace audible...

Conexiones con redes sociales

- Mar, 23/11/2010 - 1:36pm
- YouTube Petete favorited Los Tres en Raya con Kanka - Lo mal que estoy (y lo poco que me quejo) 1:36pm#
- Jue, 11/11/2010 - 2:10pm
- YouTube Petete favorited Los Tres en Raya - Promo Libertad 8 (17-11-2010) 2:10pm#
- Mié, 10/11/2010 - 2:11pm
- YouTube Petete favorited Paco Cifuentes con Petete - Platos Rotos (Sala Clamores, 27/10/2010) 2:11pm#
- Mié, 13/10/2010 - 6:52am
- YouTube Petete favorited PETETE en el Buho Real (10 de Junio de 2010) 6:52am#
- YouTube Petete favorited Petete - séptimo arte 6:51am#
- YouTube Petete favorited Petete - caleidoscopio urbano 6:51am#
- YouTube Petete favorited Petete - todo va bien 6:50am#
- YouTube Petete favorited Petete - la luz 6:50am#
- YouTube Petete favorited Petete - La luz 6:49am#
- YouTube Petete favorited Petete - Sumergido en su mundo 6:48am#

Historial

- Miembro durante**
5 semanas 5 días

Figura B.15: Perfil del músico seleccionado

La otra opción será pulsar sobre el botón 'Música/Noticias', aquí veremos una página personalizada del músico (Figura B.16) una lista para acceder a sus álbumes, un enlace para mandarle un mensaje privado (Figura B.17) y otro para ver todas las noticias que este músico haya publicado en redes sociales (Figura B.18). Los usuarios anónimos pueden acceder a esta parte de listas de músicos pero no podrán leer la información del perfil del músico ni enviarla un mensaje privado mientras no se registren.



Figura B.16: Pagina personalizada del músico



Figura B.17: Envío de un mensaje privado al músico.



Figura B.18: Últimas noticias del músico seleccionado

Podemos ver que si pulsamos sobre cualquier enlace de este músico con las redes sociales nos llevará directamente a la página que se trate, en este caso si hacemos clic sobre el primer enlace accederemos a un video de Youtube (Figura B.19).



Figura B.19: Primer enlace del músico seleccionado a la página youtube.

Estadísticas

Esta sección es para ver algunas estadísticas de OL-RADIO, por el momento podemos ver los rankings de las 5 canciones más reproducidas (son las cinco canciones que más se han seleccionado por las predicciones, nunca de forma aleatoria) vendrán ordenadas desde la primera a la quinta (Figura B.20) y podremos acceder al álbum si queremos escucharla o puntuarla (si esta registrado).

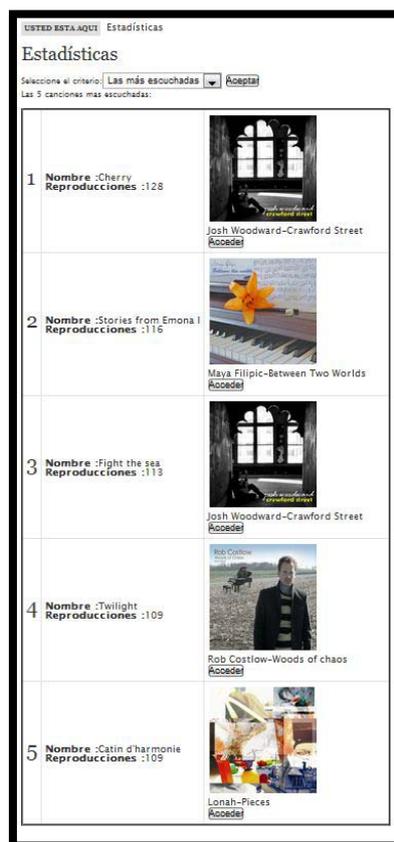


Figura B.20: Estadísticas con las 5 canciones más reproducidas.

El otro ranking disponible es el de las 5 canciones más puntuadas (Figura B.21) que se encontrarán dispuestas de forma similar a las anteriores, para acceder a él nos vamos a 'Seleccione el criterio' seleccionamos 'Las más puntuadas' y pulsamos 'Aceptar'. Los usuarios **anónimos** podrán ver sin problemas las estadísticas.



Figura B.21: Estadísticas con las 5 canciones más puntuadas.

Libro de visitas

En esta sección el usuario puede leer comentarios o firmas que otros usuarios han escrito, así como escribir el suyo propio, haciendo clic sobre 'Añadir nuevo comentario'. El usuario **anónimo** podrá leer estos comentarios pero no podrá escribir ninguno.



Figura B.22: Libro de visitas

Quiénes somos

Esta sección tendrá el único objetivo de ofrecer al usuario un conocimiento sobre OL-RADIO y las personas implicadas. El usuario **anónimo** podrá acceder a esta información sin problemas.



Figura B.23: Quiénes somos

Otras funcionalidades en el menú Radio

Debido al actual auge de las redes sociales como Twitter o Facebook, OL-Radio ofrece la posibilidad a sus usuarios de publicar en estas redes sociales las partes del sitio del menú Radio.

Conexión con Facebook

En la parte inferior de los contenidos del menú Radio veremos una opción 'Me gusta' para conectar con Facebook (Figura B.24).



Figura B.24: Enlaces de me gusta para enlazar con Facebook.

Si pulsamos esta opción nos aparecerá una ventana emergente donde debemos introducir nuestros datos de cuenta en Facebook (Figura B.25).



Figura B.25: Ventana emergente para introducir los datos de Facebook.

Automáticamente nos aparecerá el enlace de OL-RADIO en nuestro espacio en Facebook (Figura B.26).



Figura B.26: Pagina Facebook con el enlace a OL-RADIO.

Conexión con twitter

Para publicar en Twitter alguna sección de OL-RADIO, debemos pulsar el enlace que se encuentra en la parte inferior 'Publicar en Twitter'.

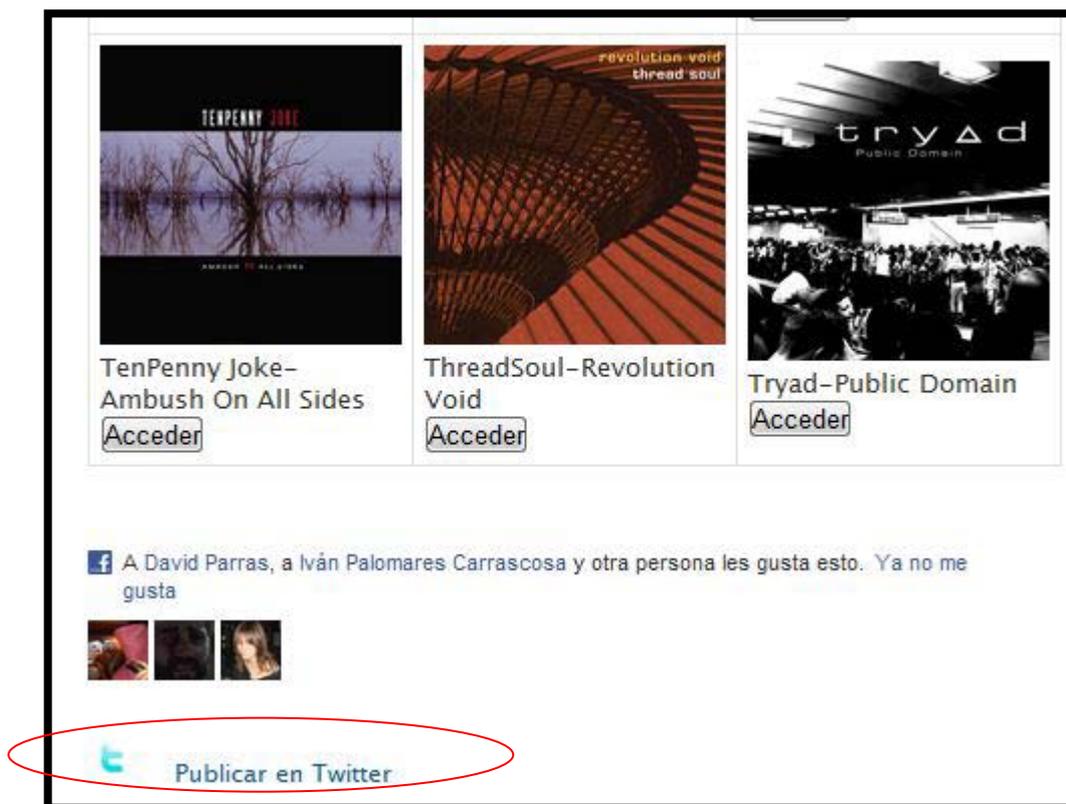


Figura B.27: Enlace para publicar un contenido en twitter.

Entonces nos aparecerá una ventana emergente para introducir nuestros datos en Twitter.

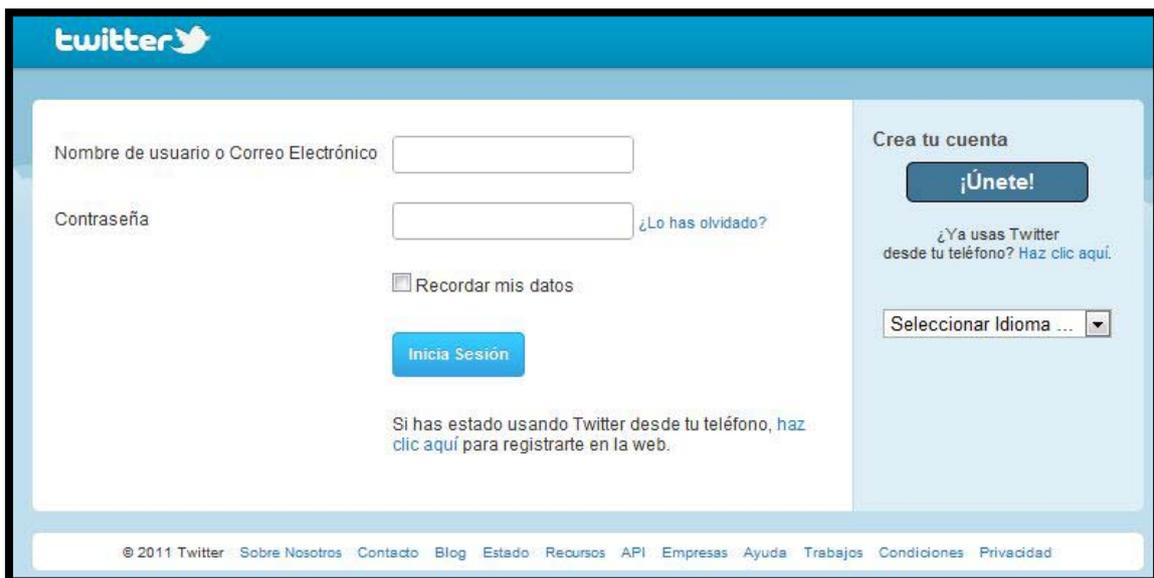


Figura B.28: Introduciendo los datos de twitter.

Si vemos nuestro perfil en Twitter ya tenemos el enlace de OL-RADIO en ella.

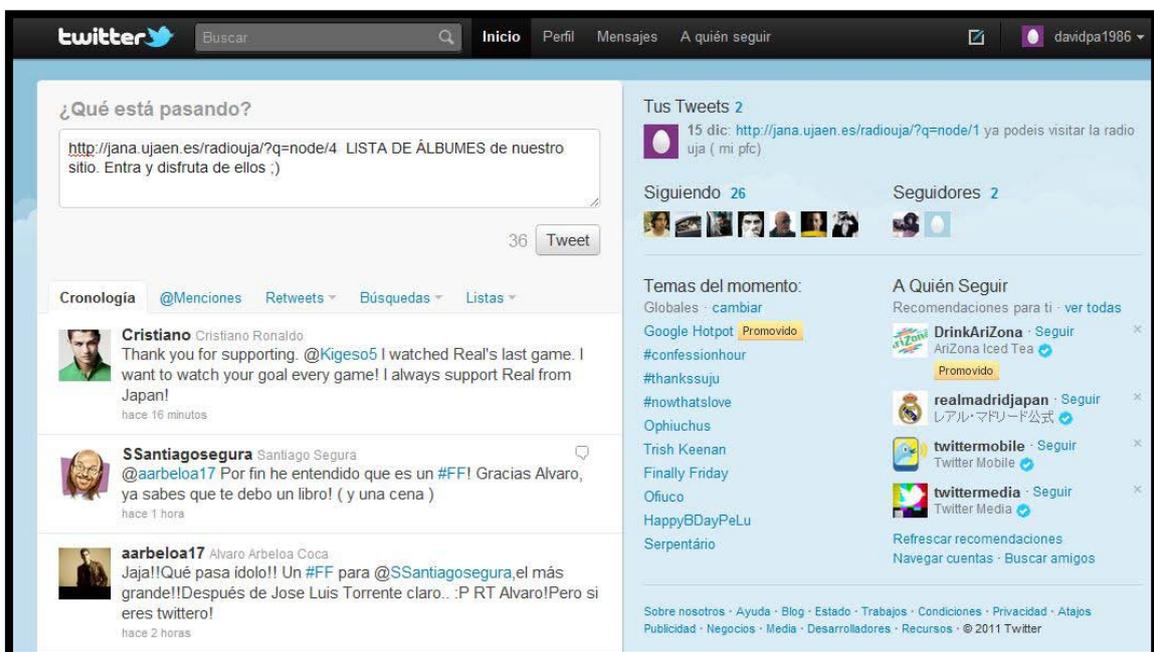


Figura B.29: Enlace de OL-RADIO en twitter.

B.3. Menú Personalizado de usuario (usuario registrado)

Ahora vamos a analizar cada una de las funciones de este menú personalizado de un usuario registrado (**oyente o músico**).

Mi cuenta

Desde aquí, el usuario puede modificar sus datos personales: dirección de correo electrónico y contraseña, además de opciones de idioma, franja horaria, subir una foto como cara virtual y cambiar su género musical favorito si lo desea. Solamente hay que hacer clic sobre la opción 'Editar' (Figura B.30), modificar el campo deseado y pulsar el botón 'Guardar'. En caso de introducir un dato erróneo, el sistema nos mostrará un mensaje de forma similar a como se explicó en el registro de usuarios.

The screenshot shows the 'oyente3' user profile editing interface. At the top, there are 'Ver' and 'Editar' buttons. Below that, there are tabs for 'Cuenta' and 'Conexiones con redes sociales'. The main content is organized into several sections:

- Información de la cuenta:** Includes a 'Dirección de correo electrónico' field with the value 'oyente@oyente.es', a 'Contraseña' field with masked characters, and a 'Confirmar contraseña' field. A note explains that the email is used for system communications and password resets.
- Imagen:** Features a 'Subir imagen' section with a 'Seleccionar archivo' button and a note that the image dimensions should be 85x85 pixels and the size should not exceed 30 kB.
- Información perfil:** Contains a 'Género musical favorito' dropdown menu currently set to 'acustica'.
- Opciones de idioma:** Shows radio buttons for 'Inglés (English)' (selected) and 'Español'.
- Opciones de localización:** Includes a 'Zona horaria' dropdown menu set to 'Jueves, January 13, 2011 - 20:20 +0100'.

A 'Guardar' button is located at the bottom left of the form.

Figura B.30: Editando perfil.

También podemos conectar nuestro perfil con algunas redes sociales, para eso pulsamos en la pestaña 'Conexiones con redes sociales' y rellenamos el campo de la red social que nos interese, normalmente será el nombre de usuario que tenga el usuario en la red social que queramos conectar (Figura B.31).

Conexiones con redes sociales

Ver Editar

Cuenta Conexiones con redes sociales

Feed configuración:

Dirección URL de Feed:

URL de los feeds a la importación como un flujo de actividad. Múltiples canales se permiten colocar cada uno en una línea separada.

Twitter configuración:

Usuario:
Pepe124
El nombre de usuario en tu cuenta de Twitter

Bienvenido a la página de configuración de la actividad corriente en redes sociales. Usted puede publicar sus noticias simplemente proporcionando alguna información de la cuenta para los sitios de terceros.

Digg configuración:

Usuario:
Tu nombre de usuario en Digg

Delicious configuración:

Usuario:
El nombre de usuario de tu cuenta en Delicious

lastfm configuración:

Usuario:
Nombre de usuario en tu cuenta de lastfm

Flickr configuración:

Usuario:
El nombre de usuario de tu cuenta en Flickr

YouTube configuración:

Usuario:
pepito
El nombre de usuario para tu cuenta de YouTube para importar tus videos favoritos

Borrar Guardar

Configuración de Facebook:

Copia la URL de tu estado RSS en Facebook :

Entra en tus notificaciones pulsa el enlace Por RSS y copia la dirección URL.

Figura B.31: Conectando con redes sociales.

En esta sección hay una peculiaridad para los usuarios de tipo **músico**, esta es que estos pueden rellenar tres campos de 'Información general' que no pueden rellenar el resto, estos son el género musical de este, su sitio web y su biografía (Figura B.32).

▼ Opciones de localización

Zona horaria:
Jueves, January 13, 2011 - 20:43 +0100 ▼
Seleccione la hora local. Las fechas y horas de todo el sitio se mostrarán conforme a esta zona horaria.

Información general

Género:
Acústica / Fusión / Rock
Describenos el género con el que más te identificas

Sitio Web:
http://www.petete-laband.com/
Escribe si dispones de el tu sitio web.

Bio:
Antonio Hernando "Petete" nace en Jaén. Desde niño muestra un gran interés por la música, escuchando vinilos paternos y más tarde, enganchándose tardes enteras a los famosos programas de radio. A los 14 empieza a componer sus primeras canciones a guitarra y armónica, compaginando más

Háblanos sobre ti :

Guardar

Figura B.32: Rellenando los campos específicos del usuario de tipo músico.

Mis puntuaciones

Esta sección es como una ficha personal de las preferencias musicales del usuario, donde aparece no sólo su género favorito, sino también el listado de la canciones que han sido evaluadas por él, ordenadas de mayor a menor puntuación.

Mis puntuaciones

Género favorito: acustica
 Cambiar Género: --ELEGIR--

Mis puntuaciones

Canción	Artista	Género	Puntuación	Modificar puntuación
Stories from Emona I	Maya Filipic	acustica	4	<input type="radio"/> ★★★★★ <input checked="" type="radio"/> ★★★★★☆ <input type="radio"/> ★★★★★☆☆ <input type="radio"/> ★★★★★☆☆☆ <input type="radio"/> ★★★★★☆☆☆☆ <input type="button" value="Modificar"/>
The mission	Josh Woodward	acustica	3	<input type="radio"/> ★★★★★☆ <input type="radio"/> ★★★★★☆☆ <input checked="" type="radio"/> ★★★★★☆☆☆ <input type="radio"/> ★★★★★☆☆☆☆ <input type="radio"/> ★★★★★☆☆☆☆☆ <input type="button" value="Modificar"/>
Stories from Emona II	Maya Filipic	acustica	1	<input type="radio"/> ★★★★★☆ <input type="radio"/> ★★★★★☆☆ <input type="radio"/> ★★★★★☆☆☆ <input type="radio"/> ★★★★★☆☆☆☆ <input checked="" type="radio"/> ★★★★★☆☆☆☆☆ <input type="button" value="Modificar"/>

Figura B.33: Mis puntuaciones

El usuario puede, en cualquier momento, acceder para modificar su género musical favorito, seleccionando un género en la lista desplegable y pulsando 'Aceptar'. También puede modificar una puntuación previamente dada a una canción, cambiando el valor numérico en la fila perteneciente a dicha canción y pulsando el correspondiente botón 'Modificar'.

Mensajes

Desde esta sección un usuario puede escribir mensajes a otros usuarios y leer los mensajes recibidos. Para escribir mensajes tendremos que escribir el nombre o nombres (separados por comas) que serán los destinatarios del mensaje (Figura B.34).

Escribir mensaje nuevo

Mensajes > Escribir mensaje nuevo

A:

Separa varios destinatarios por comas.

Asunto:

Mensaje:

- Las direcciones de las páginas web y las de correo se convierten en enlaces automáticamente.
- Etiquetas HTML permitidas: <a> <cite> <code> <dl> <dt>

- Saltos automáticos de líneas y de párrafos.

Más información sobre opciones de formato

Figura B.34: Envío de un mensaje a varios usuarios

Cuando recibamos un nuevo mensaje seremos notificados en la página de inicio con un texto y en la opción la opción Mensajes aparecerá entre paréntesis el numero de mensajes no leídos (Figura B.35).

RADIO

- Lanzar radio
- Álbumes
- Músicos
- Estadísticas
- Libro de visitas
- Quiénes somos

OYENTE3

- Mi cuenta
- Mis puntuaciones
- Mensajes (1 nuevo)
- Terminar sesión

USTED ESTA AQUI Radio

Radio

Tienes 1 mensaje sin leer

Hola oyente3, te recomendamos las siguientes canciones:

- 1 - Maya Filipic-Stories from Emona I
- 2 - Josh Woodward-Thanks for coming
- 3 - Josh Woodward-Ships
- 4 - Josh Woodward-Waiting takes time
- 5 - Josh Woodward-The mission

Información de la lista de reproducción (Pulsa [aquí](#) para cambiar la lista)

Pista: 1 Stories from Emona I **Artista:** Maya Filipic **Album:** Between Two Worlds **Género:** acustica

Figura B.35: Notificaciones de recepción de nuevo mensaje

Terminar sesión

Mediante esta opción del menú se nos permite cerrar nuestra sesión en la aplicación cada vez que queramos terminar de utilizarla.

B.4. Menú Mi Música (parte específica para usuario de tipo músico)

Este menú aparecerá solamente para los usuarios que se hayan registrado como tipo **músico**, a continuación vamos a explicar esta funcionalidad que servirá al usuario a crear sus propios álbumes y subir sus canciones.

Subir álbum

Desde aquí, el usuario podrá rellenar un formulario (Figura B.36) para la creación de un nuevo álbum. El usuario deberá escribir un título obligatoriamente que no se encuentre en la base de datos, elegir el género musical que más se adecue al álbum y la licencia Creative Commons que desee. El usuario podrá subir una imagen que se utilizará para portada del álbum, sino lo desea el sistema le asignara una foto de portada por defecto. Cuando hayamos rellenado los datos pulsamos el botón 'Enviar'.

Subir álbum

Dime el título de tu álbum :

Seleccione el género que mas se adapte a tu álbum :

- Rock
- Acústica
- Techno
- Instrumental
- Pop
- House
- Psicodelic
- Ambiental

Sube una imagen para la portada de tu álbum :

Captura.PNG

El archivo debe ser de tipo imagen(jpg,jpeg,png,gif,bmp) y tener un tamaño máximo de 5 MB.

Seleccione la licencia optuna :

<input type="radio"/>	Reconocimiento: Tu obra puede ser distribuida, copiada y exhibida, dejando claro que tú eres el autor.
<input type="radio"/>	Reconocimiento – Sin obra derivada: Tu obra puede ser distribuida, copiada y exhibida, pero no pueden crear otras obras a partir de la original.
<input checked="" type="radio"/>	Reconocimiento – Sin obra derivada – No comercial: Tu obra puede ser distribuida, copiada y exhibida, pero no pueden obtener ningún beneficio económico ni realizar otras obras a partir de la original.
<input type="radio"/>	Reconocimiento – No comercial: Tu obra puede ser distribuida, copiada y exhibida, pero no pueden obtener ningún beneficio económico.
<input type="radio"/>	Reconocimiento – No comercial – Compartir igual: Tu obra puede ser distribuida, copiada y exhibida, pero no pueden obtener ningún beneficio económico. Pueden crear otras obras a partir de la tuya siempre que lleven la misma licencia que tu obra original.
<input type="radio"/>	Reconocimiento – Compartir igual: Tu obra puede ser distribuida, copiada y exhibida. Pueden crear otras obras a partir de la tuya siempre que lleven la misma licencia que tu obra original.

Figura B.36: Formulario para la creación de un nuevo álbum.

Si hemos creado el álbum sin problemas el sistema nos informará de ello y nos permitirá la opción de ir al menú para empezar a subir canciones (Figura B.37).

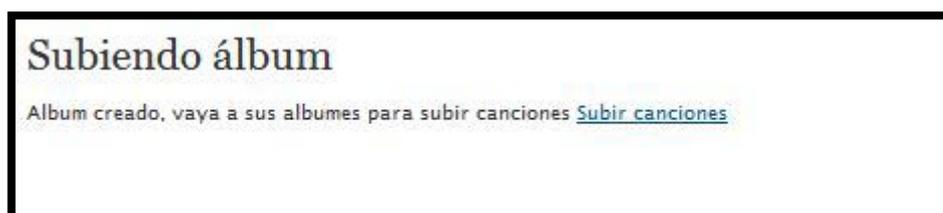


Figura B.37: Verificación de que el álbum ha sido subido sin problemas.

En caso de no introducir un título o uno que ya se encuentre el sistema nos informará de ello y nos obligará a volver al formulario para introducir un título (Figura B.38) o uno correcto (Figura B.39).

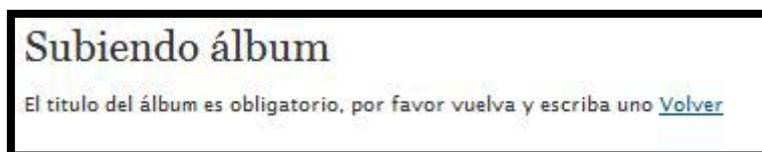


Figura B.38: Aviso del sistema de que no se ha introducido ningún título.



Figura B.39: Aviso del sistema de que el título introducido ya se encuentra en el sistema.

Con respecto a la imagen para utilizar de portada, las restricciones son que sea un archivo de tipo imagen y no superar un tamaño máximo de 5 MB. En caso de no cumplir con estas restricciones el álbum se creara sin problemas pero sin imagen, posteriormente se utilizara una por defecto.

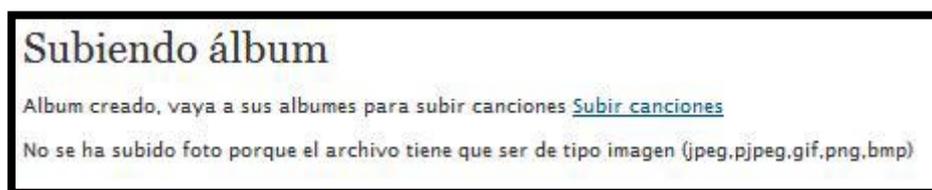


Figura B.40: El álbum se ha creado correctamente pero la imagen no era correcta.

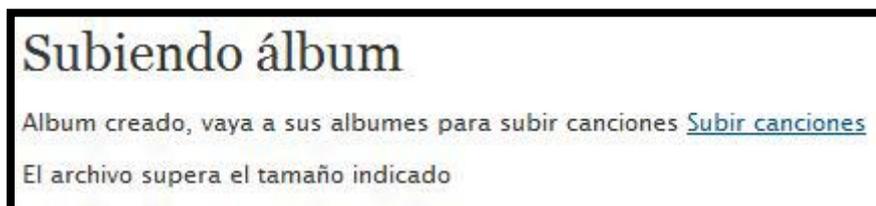


Figura B.41: El álbum se ha creado correctamente pero la imagen no cumplía con las restricciones de tamaño.

Mis álbumes

En esta sección el usuario podrá subir sus canciones a los álbumes creados, eliminar canciones subidas, editar o eliminar álbumes. Si el usuario aun no ha subido ningún álbum el sistema le informara de ello y le facilitará un enlace para empezar a subir (Figura B.42).



Figura B.42: El usuario aun no ha creado ningún álbum.

En caso de haber creado algún álbum anteriormente el sistema le mostrará una lista con sus álbumes y el usuario podrá acceder para empezar a trabajar con él pulsando sobre el botón 'Acceder' (Figura B.43).



Figura B.43: Lista de los álbumes creados por el usuario.

Si accedemos a un álbum podemos ver una breve información de este álbum, un reproductor con las canciones de este álbum, una tabla con las canciones subidas con la posibilidad de borrarlas y un pequeño formulario en la parte inferior para seguir subiendo canciones.

Mis álbumes

Seleccione un álbum para subir canción o editarlo :

Pruebas **Acceder**

Haciendo ruido **Acceder**

Opciones del álbum Haciendo ruido

Artista: Petete
Album: Haciendo ruido
Género: rock
Número canciones: 5
Tipo de licencia: Reconocimiento - Sin obra derivada - No comercial

Eliminar Album **Editar Album**

1 - Petete-**Todo va bien**
2 - Petete-La canción contra los jefes
3 - Petete-La luz
4 - Petete-Septimo arte

Canciones:

1. Petete- Todo va bien	Eliminar
2. Petete-La canción contra los jefes	Eliminar
3. Petete-La luz	Eliminar
4. Petete-Septimo arte	Eliminar
5. Petete-Vivir adrede	Eliminar

Sube una canción:

Dime el título de tu canción

Sube el archivo mp3 No se ha... archivo

El archivo tiene que ser tipo mp3 y tener un tamaño máximo de 10MB.

Enviar

Figura B.44: Información sobre el álbum seleccionado.

Ahora vamos a explicar toda la funcionalidad que nos ofrece esta sección. Una de ellas es poder eliminar el álbum completamente con todo su contenido, para ello tenemos que pulsar el botón 'Eliminar Álbum' en el cuadro de información del álbum. El sistema nos mostrara una pantalla de confirmación (Figura B.45) en la que el usuario puede cancelar la operación pulsando 'Cancelar' ya que esta acción no se podrá

deshacer, en el caso de que usuario quiera eliminar el álbum de todos modos pulsará el botón 'Eliminar'.



Figura B.45: Confirmación para la eliminación de un álbum.

Si el usuario desea editar algunos de los aspectos del álbum deberá pulsar el botón 'Editar Álbum' que se encuentra dentro del cuadro de información del álbum. El sistema mostrará un formulario para el álbum similar a cuando lo creo, con la diferencia que ahora estarán marcadas las opciones que tiene el álbum actualmente. Si quiere cambiar la imagen de portada solamente tendrá que subir una nueva imagen (en caso de no introducir ninguna o introducir una incorrecta se seguirá utilizando la misma imagen que antes). Para realizar los cambios debemos pulsar el botón 'Cambiar'. Las restricciones serán las mismas que para crear álbum, es decir se deberán introducir un nombre que no esté registrado anteriormente y un archivo de tipo imagen correcta.

Editar/Eliminar álbum

Artista: Petete
Album: Haciendo ruido
Género: rock
Número canciones: 5
Tipo de licencia: Reconocimiento - Sin obra derivada - No comercial



Dime el título de tu álbum :

Seleccione el género que mas se adapte a tu álbum :

- Rock
- Acústica
- Techno
- Instrumental
- Pop
- House
- Psicodelic
- Ambiental

Sube una imagen para cambiar la portada de tu álbum :

No se h...archivo

El archivo debe ser de tipo Imagen(jpg, jpeg, png, gif, bmp) y tener un tamaño máximo de 5 MB.

Seleccione la licencia oportuna :

<input type="radio"/> 	Reconocimiento: Tu obra puede ser distribuida, copiada y exhibida, dejando claro que tú eres el autor.
<input type="radio"/> 	Reconocimiento - Sin obra derivada: Tu obra puede ser distribuida, copiada y exhibida, pero no pueden crear otras obras a partir de la original.
<input checked="" type="radio"/> 	Reconocimiento - Sin obra derivada - No comercial: Tu obra puede ser distribuida, copiada y exhibida, pero no pueden obtener ningún beneficio económico ni realizar otras obras a partir de la original.
<input type="radio"/> 	Reconocimiento - No comercial: Tu obra puede ser distribuida, copiada y exhibida, pero no pueden obtener ningún beneficio económico.
<input type="radio"/> 	Reconocimiento - No comercial - Compartir igual: Tu obra puede ser distribuida, copiada y exhibida, pero no pueden obtener ningún beneficio económico. Pueden crear otras obras a partir de la tuya siempre que lleven la misma licencia que tu obra original.
<input type="radio"/> 	Reconocimiento - Compartir igual: Tu obra puede ser distribuida, copiada y exhibida. Pueden crear otras obras a partir de la tuya siempre que lleven la misma licencia que tu obra original.

Editando álbum

El archivo debe de ser de tipo imagen y no superar el tamaño indicado
¡Cambios efectuados!

Figura B.46: Formulario para editar el álbum y mensaje de cambios efectuados.

Con respecto a las canciones podremos escucharlas, tan solo pulsando el botón 'play' en el reproductor, o eliminarlas con solo pulsar el botón 'Eliminar' en la canción

que queramos eliminar. El sistema nos mostrará un mensaje de confirmación para que el usuario acepte o cancele la operación. Si desea cancelar la operación deberá pulsar 'Cancelar' para eliminar pulsar el botón 'Eliminar'.



Figura B.47: Mensaje de confirmación para eliminar canción.

Para acabar si el usuario desea subir una nueva canción bastará con rellenar el formulario de la parte inferior 'Sube una canción' y rellenar los datos con un nombre para la canción, un archivo mp3 y pulsar el botón 'Enviar'.

Figura B.48: Formulario para la subida de canciones.

Si la subida ha sido incorrecta el sistema nos informará de ello (Figura B.49).

Figura B.49: Errores en la subida de canción.

Si la subida al servidor ha sido correcta el sistema nos lo confirmará y nos informará de que la canción será subida a la radio cuando la revise un usuario administrador (Figura B.50).



Figura B.50: Confirmación de subida de canción.

B.5. Menú Administrar Radio (parte específica para usuario de tipo administrador radio)

Debido a la ampliación de funcionalidades y la posibilidad por parte de los usuarios de la subida de música nos vemos obligados a crear la figura de un **administrador radio** que se encargará del buen funcionamiento del sitio. A continuación vamos a explicar con todo lujo de detalles las funciones específicas de este usuario.

Revisar canciones

Cada vez que un usuario de tipo **músico** sube una nueva canción, esta automáticamente pasará a formar parte de la lista de canciones pendientes de revisar. En esta lista un usuario **administrador radio** podrá marcar las canciones que desea subir y pulsar el botón 'Subir' o marcar canciones que no deseas que sean subidas y pulsar el botón 'Eliminar' (Figura B.51).

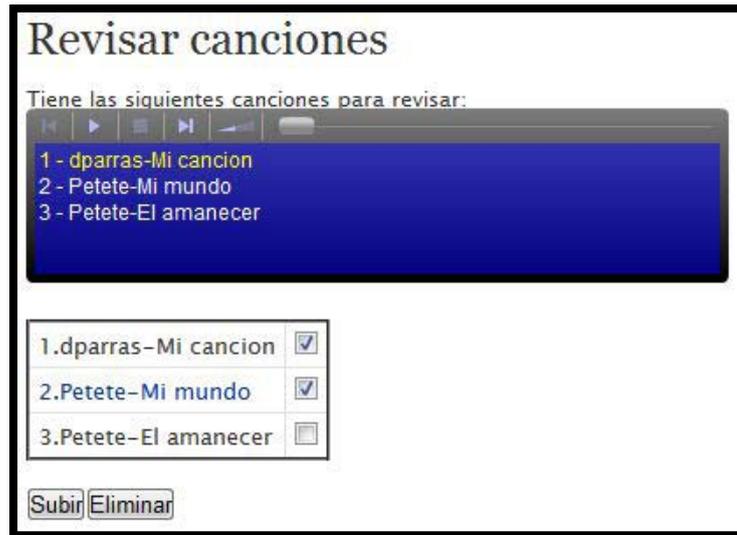


Figura B.51: Marcando canciones para subir o eliminar.

Cuando no haya más canciones para subir el sistema nos informará de ello (Figura B.52).



Figura B.52: No hay más canciones para revisar.

Ejecutar algoritmo

Desde aquí, el usuario puede hacer dos funciones. La primera es ejecutar manualmente el algoritmo de filtrado, el cual actualizará las predicciones y recomendaciones a los usuarios basándose en sus puntuaciones. Para ello nos fijamos en la parte inferior de la pantalla 'Ejecución manual' como podemos observar el sistema dará información sobre la última ejecución (ya que puede ser una acción costosa) y para ejecutar el algoritmo tan solo debemos pulsar el botón 'Ejecutar Algoritmo' que está situado en el apartado "ejecución manual" (Figura B.53).

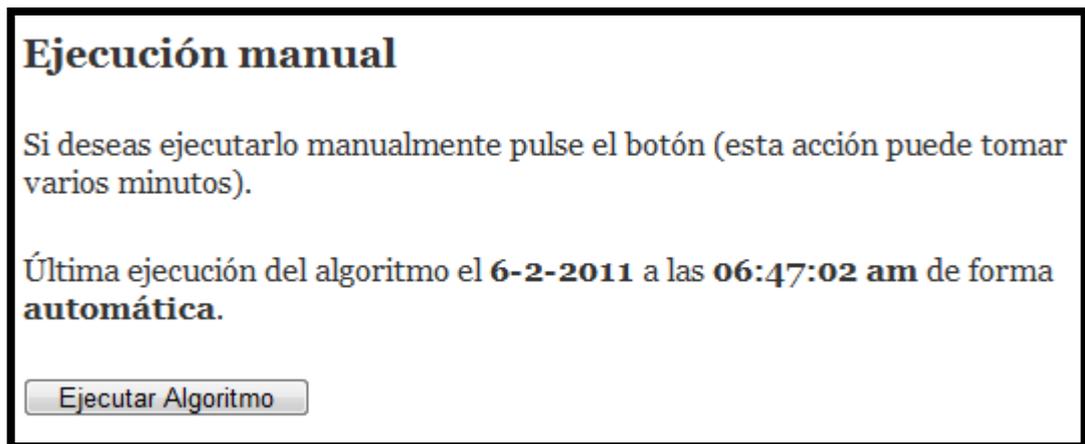


Figura B.53: Ejecución manual.

Si el algoritmo se ejecutó sin problemas el sistema nos informará de ello.



Figura B.54: Información para saber que todo ha ido correctamente.

La segunda opción es configurar la parte de ejecuciones automáticas del algoritmo. Para ello en la parte superior de la pantalla "ejecución automática" (Figura B.55) encontraremos un breve formulario donde podemos indicar la hora con los minutos exactos y el día que queremos para la ejecución (también tenemos la opción de diariamente); una vez realizada la configuración pulsamos el botón 'Cambiar'.

Ejecutar algoritmo

Ejecución automática

El algoritmo en estos momentos se ejecuta todos los sábados a las 6:25.

Si desea cambiarlo, rellene los siguientes campos según sus preferencias.

Hora: 9 Minuto: 12 Día: Sábado

Ejecución manual

Si deseas ejecutarlo manualmente, pulsa el botón (esta acción puede tomar varios minutos).

- Diariamente
- Lunes
- Martes
- Miercoles
- Jueves
- Viernes
- Sábado
- Domingo

Figura B.55: Configuración de ejecución automática.

El sistema nos devolverá a la misma pantalla pero podremos ver que la configuración ha cambiado según nuestras preferencias (Figura B.56).

Ejecutar algoritmo

Ejecución automática

El algoritmo en estos momentos se ejecuta diariamente a las 9:12.

Si desea cambiarlo, rellene los siguientes campos según sus preferencias.

Hora: 9 Minuto: 12 Día: Diariamente

Figura B.56: Nueva configuración de ejecución manual.

Eliminar canciones

El usuario administrador radio también puede eliminar una canción que se encuentre subida en el sitio, ya que puede que el momento de ser revisada no lo hizo correctamente o no se percató de un posible error, por tanto en esta sección tendremos que seleccionar el álbum (pulsando 'Acceder') al que pertenece la canción.



Figura B.57: Lista de álbumes para eliminar canciones.

Una vez seleccionado el álbum marcar las canciones que queremos eliminar y pulsar el botón 'Eliminar' (Figura B.58), como es habitual podemos comprobar las canciones que queremos eliminar escuchándolas primero o pulsar el enlace 'volver' para regresar a la lista generales de álbumes.

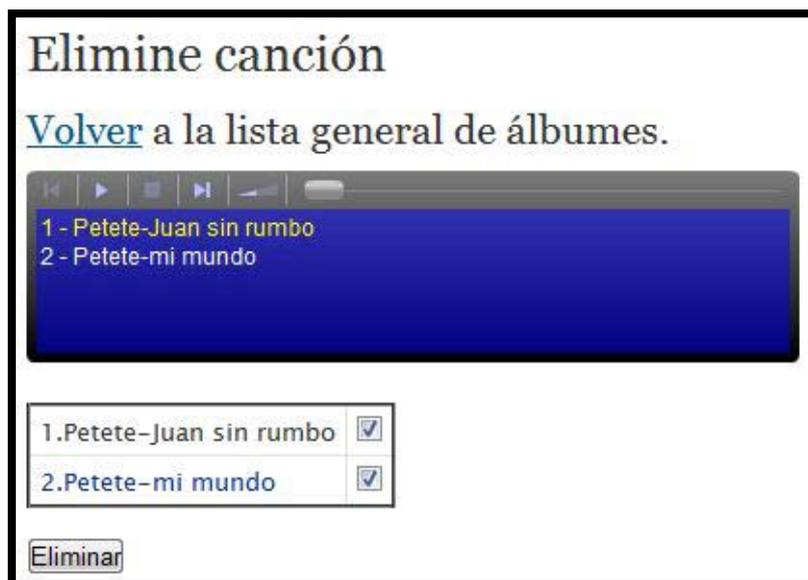


Figura B.58: Marcando las canciones para eliminar.

Eliminar álbum

Para acabar, si un usuario desea eliminar un álbum entero con todo su contenido también puede hacerlo, tan solo tendrá que elegir el álbum de la lista de álbumes a eliminar pulsar sobre el botón 'Eliminar' (Figura B.59).



Figura B.59: Seleccionando un álbum para eliminar.

A continuación confirmar la acción que desea hacer, si decide eliminar de todas formas pulsar 'Eliminar' para cancelar pulsar 'Cancelar' (Figura B.60).



Figura B.60: Confirmación de eliminación de un álbum.

ANEXO C

Código fuente del Algoritmo de filtrado colaborativo

En este tercer anexo incluimos la implementación en lenguaje Java del algoritmo de filtrado colaborativo que se desarrolló para el anterior proyecto y que para este hemos tenido que incluir alguna pequeña modificación. Se trata de un algoritmo basado en ítem, del tipo 'vecino más cercano' (K-nn), que emplea el coeficiente coseno como medida de similitud entre canciones, y el algoritmo weighted-sum para calcular las predicciones de los usuarios sobre las canciones.

Hemos considerado útil dedicar un anexo a este código fuente, ya que consideramos que de esta forma el lector interesado podrá encontrar una cómoda referencia a la hora de implementar un algoritmo de filtrado colaborativo o, simplemente, para aquel lector que desee profundizar en el funcionamiento de dicho algoritmo.

El algoritmo se mantiene igual que en el anterior proyecto [28] con tan solo la novedad de la modificación en el punto C1, donde esta vez tenemos que tener en cuenta las canciones que se borrarán lógicamente o que aun no han sido revisadas.

C.1. Contenido del fichero Main.java

```
package algoritmofiltrado;
import java.sql.*;
import java.util.Vector;
import java.math.*;

/**
 * La clase Main es la clase principal o clase de Aplicación del algoritmo. Incluye el
 * establecimiento de la conexión con la base de datos y las llamadas a los distintos
 * métodos para llevar a cabo el algoritmo de filtrado.
 */
public class Main {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {

        //Preparación de las clases de la API JDBC para conectar con la Base de Datos, en esta
        try {
            Class.forName("org.gjt.mm.mysql.Driver");
            //Nueva conexión con el servidor jana
            Connection conexion =
                DriverManager.getConnection("jdbc:mysql://127.0.0.1:3306/pfc-radio", "pfc-
                radio", "contraseña_en_el_servidor");
            System.out.println("Conexion realizada");
            //Ejecución de una sentencia SQL: consulta de todas las canciones de la BD.
            //Volcado de los identificadores de las canciones en un vector.
            Statement s = conexion.createStatement();
            Vector<Integer> id_canciones = new Vector<Integer>();
            //Solo nos interesan las canciones que no esten borradas y que hayan sido revisadas
            ResultSet rs = s.executeQuery("select id_cancion from canciones WHERE
            borrado='no' and revisada='si'");
            while (rs.next()) {
                id_canciones.add(rs.getInt(1));
            }
            //Para cada par de canciones, se calcula la similitud entre ellas.
            //Guardamos, para cada canción, las similitudes con el resto de canciones.
            //De esta forma podremos, mas adelante, seleccionar los k vecinos más cercanos a cada objeto.
```

```

Vector<Vector<Double>> similitudes = new Vector<Vector<Double>>();
for (int i = 0; i < id_canciones.size(); i++) {
    Vector<Double> sims = new Vector<Double>();
    for (int j = 0; j < id_canciones.size(); j++) {
        //Llamada al método coeficiente_coseno de la clase Similitud
        double sim = Similitud.coeficiente_coseno(id_canciones.get(i), id_canciones.get(j), conexion);
        System.out.println("sim(" + id_canciones.get(i) + "," + id_canciones.get(j) + ") " +
            "=" + sim);
        sims.add(sim);
    }
    similitudes.add(sims);
}

//Realizar las predicciones de cada usuario, en base a los k vecinos más cercanos a cada canción
//Inserción a actualización de la prediccion usuario-cancion en la BBDD
Statement s2 = null;
ResultSet rs2 = null;
s2 = conexion.createStatement();
rs2 = s.executeQuery("select uid from users");
while (rs2.next()) { //PARA CADA USUARIO
    int uid = rs2.getInt(1);
    Statement su = null;
    ResultSet rsu = null;
    su = conexion.createStatement();
    rsu = su.executeQuery("select value from profile_values where fid='2' and uid='" + uid + "'");
    String genero = null;
    if (rsu.first()) {
        genero = rsu.getString(1);
    } else {
        genero = "rock";
    }
    for (int x = 0; x < id_canciones.size(); x++) { //PARA CADA CANCION
        int idc = id_canciones.get(x);
        Vector<Double> aux = similitudes.get(x);
        //Llamada al método Weighted_sum de la clase Predicción
        double pred = Prediccion.weighted_sum(uid, idc, conexion, aux, id_canciones, 10);
        Statement su2 = null;
        ResultSet rsu2 = null;

```

```
        su2 = conexion.createStatement();
        rsu2 = su2.executeQuery("select genero from canciones where id_cancion=" + idc + "");
        if (rsu2.first()) {
            String genero2 = rsu2.getString(1);
            if (genero.equals(genero2)) {
                pred += 0.25;
            } //Plus de 0.25 por ser del genero favorito del usuario
        }
        su2.close();
        rsu2.close();
        System.out.println("PREDICCION DE USUARIO " + uid + " SOBRE CANCION " + idc + ": " +
pred);

        Statement ss = null;
        ResultSet rss = null;
        ss = conexion.createStatement();
        rss = ss.executeQuery("select prediccion from predicciones where id_usuario=" + uid +
            "" and id_cancion=" + idc + "");
        if (rss.first()) {
            //Ya existia Pred(u,i) en la BBDD: ACTUALIZACIÓN
            Statement stmt = conexion.createStatement();
            stmt.executeUpdate("UPDATE predicciones SET prediccion=" + pred + " WHERE
id_usuario=" + uid +
            "" and id_cancion=" + idc + "");
            stmt.close();
        } else {
            //Pred(u,i) no existia: INSERCIÓN
            PreparedStatement stmt = conexion.prepareStatement("INSERT INTO predicciones VALUES
(?,?,?)");
            stmt.setInt(1, uid);
            stmt.setInt(2, idc);
            stmt.setDouble(3, pred);
            stmt.executeUpdate();
            stmt.close();
        }
        ss.close();
        rss.close();
    }
    su.close();
    rsu.close();
}
```

```
s2.close();
rs2.close();
conexion.close();
} catch (Exception e) {
    e.printStackTrace();
}
}
}
```

C.2. Contenido del fichero Similitud.java

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package algoritmofiltrado;

import java.sql.*;

/**
 * Esta clase contiene la operación que calcula la similitud entre dos canciones, por
 * medio del coeficiente coseno
 */
public class Similitud {
    /**
     * Cálculo de la similitud entre 2 canciones con el coeficiente coseno
     * @param cancionx Identificador de la primera de las 2 canciones
     * @param canciony Identificador de la segunda de las 2 canciones
     * @param conexion La variable de conexión MySQL
     * @return El valor de la similitud, un número real entre 0 y 1.
     */
    public static double coeficiente_coseno(int cancionx, int canciony, Connection conexion) {
        try {
            Statement s = null;
            ResultSet rs = null;
            s = conexion.createStatement();
            rs = s.executeQuery("select uid from users");
            int numerador = 0;
            int denominX = 0;
            int denominY = 0;
            //PARA CADA USUARIO
            while (rs.next()) {
                int Ui = rs.getInt(1);
                Statement s2 = null;
                Statement s3 = null;
                ResultSet rs2 = null;
                ResultSet rs3 = null;
                s2 = conexion.createStatement();
```

```
//Ver si el usuario "Ui2 ha evaluado la cancion x
rs2 = s2.executeQuery("select puntuacion from puntuaciones where id_usuario="+Ui+" and " +
    "id_cancion="+cancionx+"");
s3 = conexion.createStatement();
//Ver si el usuario "Ui2 ha evaluado la cancion y
rs3 = s3.executeQuery("select puntuacion from puntuaciones where id_usuario="+Ui+" and " +
    "id_cancion="+canciony+"");
//Si el usuario actual "Ui" ha evaluado ambos items, x e y
if(rs2.first() && rs3.first()){
    //Se tiene en cuenta en el calculo del coeficiente coseno
    int Xi = rs2.getInt(1);
    int Yi = rs3.getInt(1);
    numerador += Xi*Yi;
    denominX += (Xi*Xi);
    denominY += (Yi*Yi);
}
s2.close();
s3.close();
rs2.close();
rs3.close();

}
s.close();
rs.close();
double denominador = Math.sqrt(denominX*denominY);
if (denominador==0) return 0;
return numerador/denominador;

} catch (Exception e) {
    e.printStackTrace();
}
return 0;
}
}
```

C.3. Contenido del fichero Predicción.java

```

/**
 * Esta clase incluye el método que realiza la predicción para un usuario de una canción,
 * mediante la medida llamada weighted_sum
 */
public class Prediccion {
    /**
     * Calcula la predicción mediante la medida de weighted_sum
     * @param Ua El usuario sobre el cual se realiza la predicción
     * @param Ia El ítem sobre el que se realiza la predicción
     * @param conexion La variable de conexión a MySQL
     * @param simIa Conjunto de similitudes del ítem Ia con el resto
     * @param id_canciones Identificadores de las canciones similares
     * @param k El número de vecinos más cercanos a la canción Ia que se deben tener en cuenta
     * @return El valor de la predicción, un real que se mueve inicialmente entre 0 y 5
     */
    public static double weighted_sum(int Ua, int Ia, Connection conexion, Vector<Double> simIa,
        Vector<Integer> id_canciones, int k) {
        Vector<Double> sim_Ia = new Vector<Double>(simIa);
        Vector<Double> mayor_sims = new Vector<Double>();
        Vector<Integer> mayor_items = new Vector<Integer>();
        double denominador = 0;
        double numerador = 0;
        //PASO 1: OBTENER LOS IDs DE LOS K ITEMS MAS SIMILARES A Ia, Y LAS PROPIAS
        SIMILITUDES
        for (int i = 0; i < k; i++) { //se extraen 50 similitudes
            double mayorsim = 0;
            int mayorpos = 0;
            for (int j = 0; j < sim_Ia.size(); j++) { //para extraer cada una se recorre el vector en busca del mayor
                if (sim_Ia.get(j) >= mayorsim) {
                    mayorsim = sim_Ia.get(j);
                    mayorpos = j;
                }
            }
            sim_Ia.set(mayorpos, -1.0); //se limpia la posición del mayor valor encontrado esta vez
            //System.out.println("Cancion " + id_canciones.get(mayorpos) + " : " + mayorsim);
            mayor_sims.add(mayorsim);
        }
    }
}

```

```

        mayor_items.add(id_canciones.get(mayorpos)); //en mayor_items ya se guardan los IDs de los
items
    }
//FIN PASO 1
//PASO 2: CALCULAR EL NUMERADOR Y EL DENOMINADOR
try {
    for (int i = 0; i < k; i++) {
        Statement s = null;
        ResultSet rs = null;
        s = conexion.createStatement();
        rs = s.executeQuery("select puntuacion from puntuaciones where id_usuario=" + Ua + " and " +
            "id_cancion=" + mayor_items.get(i) + "");
        //Se obtiene asi la puntuacion R(Ua,Ih)
        if(rs.first()){ //Si el item Ih ha sido puntuado por el usuario, se tiene en cuenta
            //System.out.println("cancion " + mayor_items.get(i) + " puntuada por el usuario: " +
rs.getInt(1));
            numerador += (mayor_sims.get(i)*rs.getInt(1)); //SUM[sim(Ia,Ih)*RU(Ua,Ih)]
            denominador += mayor_sims.get(i); //SUM[ | sim(Ia,Ih) | ]
        }
        s.close();
        rs.close();
    }
    //System.out.println("NUMERADOR: " + numerador);
    //System.out.println("DENOMINADOR: " + denominador);
    if(denominador==0) return 0;
    else return (numerador/denominador);
} catch (Exception e) {
    e.printStackTrace();
}
return 0;
}
}

```

Bibliografía y Referencias

- [1]Balabanovic, M. y Shoham, Y. (1997), "*Content-based, collaborative recommendation*". Communications of the ACM, Vol. 40
- [2]Cho, Y.H., J. Kyeong Kim, "A personalized recommender system based on Web usage mining and decision tree induction" *Expert Systems with Applications, Volume 23, Issue 3, 1 October 2002, Pages 329-342*
- [3]Herlocker, J., Konstan, J., Terveen, L., Riedl, J. (2004), "*Evaluating Collaborative Filtering Recommender Systems*". ACM Vol. Transactions of Information Systems, vol. 22, pp. 5-53.
- [4]Breese, J., Heckerman, D., Kadie, C., "*Empirical Analysis of Predictive Algorithms for Collaborative Filtering*", Technical Report MSR-TR-98-12, May 2008.
- [5]Vozalis, E., Margaritis, K., "*Analysis of Recommender Systems Algorithms*".
- [6] Jung, K.Y., Choi, J.H., Rim, K.W., Lee, J.H., "*Development of Design Recommender System Using Collaborative Filtering*", Digital libraries: technology and management of indigenous knowledge for global access , Vol. 2911, 2003.
- [7] Yu, K., Xu, X., Tao, J., Ester, M., Kriegel H.P., "*Instance Selection Techniques for Memory-Based Collaborative Filtering*", 12th international workshop on database and expert systems applications, proceedings, 2001.
- [8] Papagelis, M., Plexousakis, D., Rousidis, I., Theoharopoulos, E., "*Qualitative Analysis of User-based and Item-based Algorithms for Recommendation Systems*", Institute of Computer Science, Foundation for Research and Technology – Hellas.
- [9] Red social: (http://es.wikipedia.org/wiki/Red_social).

- [10] Shardanand, U., *"Social Information Filtering for Music Recommendation"*, Massachusetts Institute of Technology, Sept. 1994.
- [11] Dawson, C. W. y Martin, G. (2002), *"El proyecto fin de carrera en Ingeniería Informática: Una guía para el estudiante"* Ed. Prentice Hall
- [12] Gutierrez, A., Bravo, G. *"PHP 5 a través de ejemplos"*, Ed. RA-MA
- [13] Krug, S. *"No me hagas pensar: una aproximación a la Usabilidad en la Web"*. Ed. Prentice Hall
- [14] MySpace: (<http://es.wikipedia.org/wiki/Myspace>).
- [15] Brett D. McLaughli, Pollice, G., West, D. *"Head First Object-Oriented Analysis and Design"*. Ed. O'Reilly
- [16] Reynolds, P.A., Mason, R., Eaton, K.A., *"Webcasting: casting the Web more widely"*, British Dental Journal, Vo. 204, Issue 3, pp 145-149, 2008.
- [17] López, H., *"Multimedios en el Web: Streaming media"*, Universidad de Las Américas Puebla,
(http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/lopez_a_h/capitulo3.pdf)
- [18] Creative Commons España, (<http://es.creativecommons.org/>)
- [19] Conde, F., *"Interacción Persona-Ordenador: ¿Pero qué narices es una buena interfaz de usuario"*. Apuntes de la asignatura.
- [20] Facebook: (<http://es.wikipedia.org/wiki/Facebook>).
- [21] Twitter: (<http://es.wikipedia.org/wiki/Twitter>).
- [22] Michael Peacock *"Drupal 6 Social Networking, Build a social or community web site with friends lists, groups, custom user profiles, and much more"*, Packt Publishing.

- [23] Araque Ibáñez, A. (2008) *“Sistema de Recomendación de restaurantes Georreferenciados”*. Proyecto Fin de Carrera. Escuela Politécnica Superior (Universidad de Jaén).
- [24] Feito Higuera, F., Ruiz de Miras, J., Molina Aguilar, A. (1996), *“Análisis y Gestión de Datos”*, Editorial Universidad de Jaén.
- [25] La nueva economía: (<http://lanuevaeconomia.com>).
- [26] Experian Hitwise: (<http://www.hitwise.com/us/>).
- [27] CopyLeft : (<http://fundacioncopyleft.org/>).
- [28] Iván Palomares Carrascosa (2009) *“Radio Online basada en un motor de filtrado colaborativo”*. Proyecto Fin de Carrera. Escuela Politécnica Superior (Universidad de Jaén).
- [29] Milenio Online: (<http://impreso.milenio.com/node/8669643>).
- [30] Arranque en frío:
(http://www.territorioscuola.com/wikipedia/es.wikipedia.php?title=Cold_start).
- [31] Iván Palomares Carrascosa (2009) *“Radio Online basada en un motor de filtrado colaborativo”*. Proyecto Fin de Carrera. Escuela Politécnica Superior (Universidad de Jaén).
Capítulo 4, apartado 4.5.4 Actualización del algoritmo de filtrado, pp. 138.