



SPOTIFY

Práctica 2 Ingeniería Web

Grupo

Javier Molpeceres Gómez
Jorge Segovia Tormo

Indice

Contenido

Indice.....	1
SPOTIFY	2
Wireframes y Mockups	3
Diagramas.....	13
Descripción técnica	14
Detalles de implementación	14
Problemas encontrados	15
Mejoras y ampliaciones.....	16
Herramientas Adicionales	17
Referencias.....	18

SPOTIFY

El sistema elegido para esta para esta práctica es SPOTIFY, es un sistema de reproducción de música. Para esta práctica desarrollaremos ciertos aspectos de este sistema intentando imitar su funcionamiento. Para ello el profesor nos presenta una propuesta de las partes a desarrollar.

La propuesta del profesor fue:

Publica: registro, buscar canciones/detalle de canción, oír previews.

Privada: login, buscar canciones/detalle de canción, escucharlas, listas de reproducción.

BO: CRUD canciones, estilos, artistas.

Dejando lo de tipos e intereses de usuarios, tipos de listas y la publicidad.

Tras la reunión de planificación con el profesor y exponer los posibles inconvenientes de desarrollo sobre su propuesta por el hecho de que Spotify no tuviera una parte pública y que los estilos pertenecían a las canciones los requisitos evolucionaron hasta ser así:

Publica: login, registro.

Privada: buscar canciones/detalle de canción, escucharlas(simulado), listas de reproducción, artistas, detalle de artista.

BO: CRUD canciones, artistas, usuarios, álbumes.

Pasaremos a mostrar los elementos desarrollados:

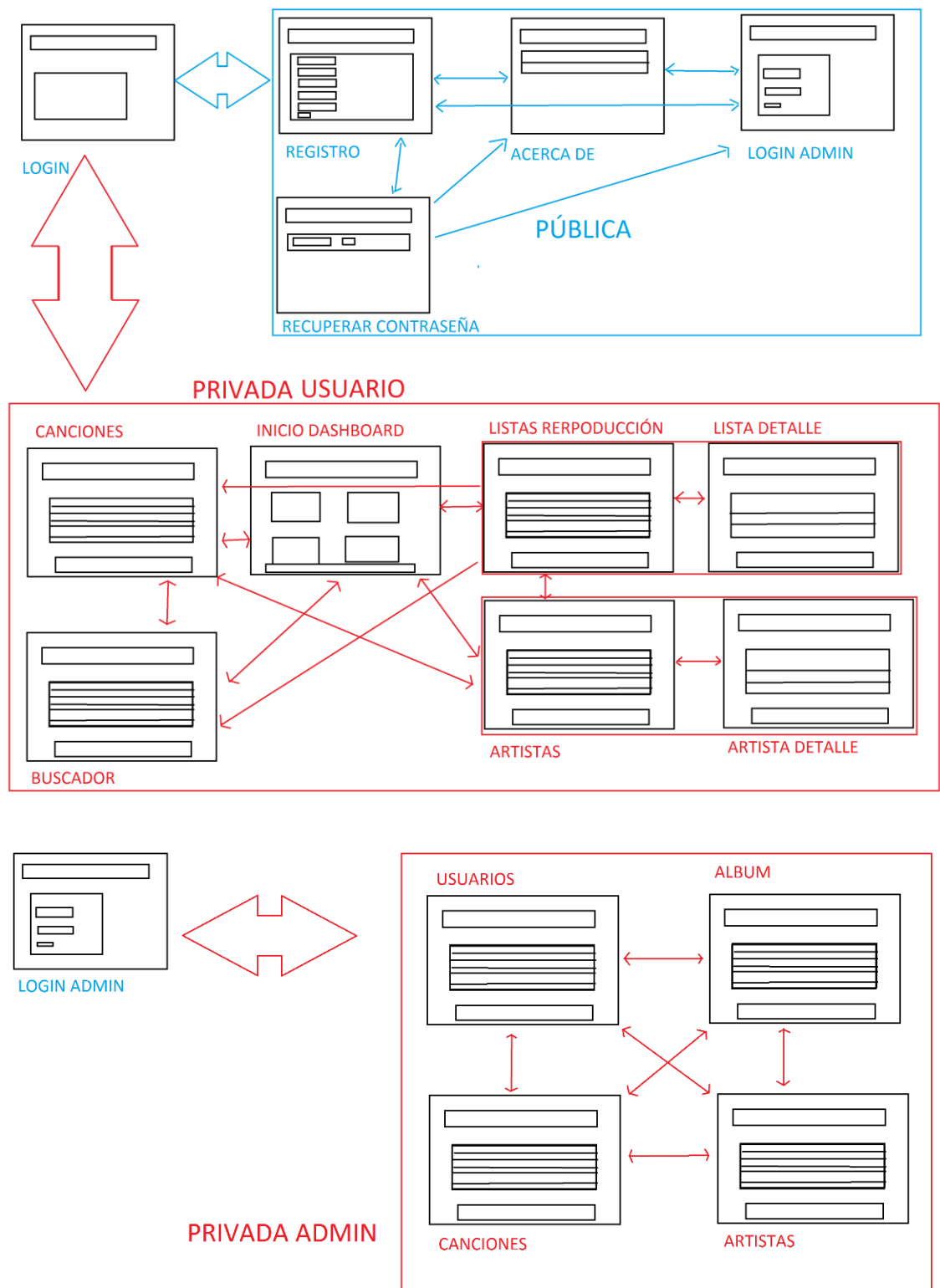
En primer lugar, la parte pública hemos desarrollado el login de usuarios de la aplicación, el login de Administración, Registro, recuperar contraseña y un “acercade” de la aplicación.

En la parte privada hemos realizado dos partes independientes. Una para administradores y otras para usuarios.

La parte privada de administrador incluye CRUD de Usuarios, Canciones, Artistas y álbumes. En la parte privada de usuario encontramos un menú de navegación(dashboard) como página de inicio, Listas de reproducción del usuario, una lista de canciones paginadas, una lista de artistas paginado y un buscador de canciones. Se puede acceder al detalle de artista, donde se verán sus detalles y sus canciones, al detalle de una canción en modo de reproducción.

El modo de reproducción es un pie de página donde se podrá ver los detalles de las canciones.

Wireframes y Mockups



Bienvenidos

[Inicio](#) [Registro](#) [Acerca de](#) [Administracion](#)

Login

Email

Contraseña

☒ Recordar

[Registrarse](#)[Olvidó la contraseña](#)

Esta es la página principal e inicial de la web, podemos ver que la acción principal es la de login donde el usuario podrá acceder como usuario normal. También se ofrece un menú de navegación superior para acceder a las principales acciones públicas que se pueden realizar sin estar logueado.

Esta pantalla ha sido diseñada con el patrón principal de identificación de usuarios, también se ha utilizado el patrón de navegación para la barra superior.

Es importante remarcar que podemos encontrar también un login especial que es solo para el administrador, aunque es igual que el login anterior pero da acceso a la parte de administrador y solo pueden acceder un usuario que tenga permisos de administrador.

Registro

[Inicio](#) [Registro](#) [Acerca de](#) [Administracion](#)

Nombre

Apellidos

Usuario

Email

Repetir Email

Contraseña

Repetir Contraseña

Ciudad

Calle

Cod Postal

Sexo

☒ Mujer ☐ Hombre

Fecha nacimiento

Nacionalidad

Options



☒ Aceptar terminos

Enviar

[Recuperar Contraseña](#)

[Volver al login](#)

En esta pantalla se ofrece a los usuarios no logueados la posibilidad de registrarse rellenando los distintos campos correctamente.

El patrón utilizado para esta pantalla es un patrón de formulario dado que ofrecemos pistas y sugerencias en los campos de texto, mensajes de error etc.

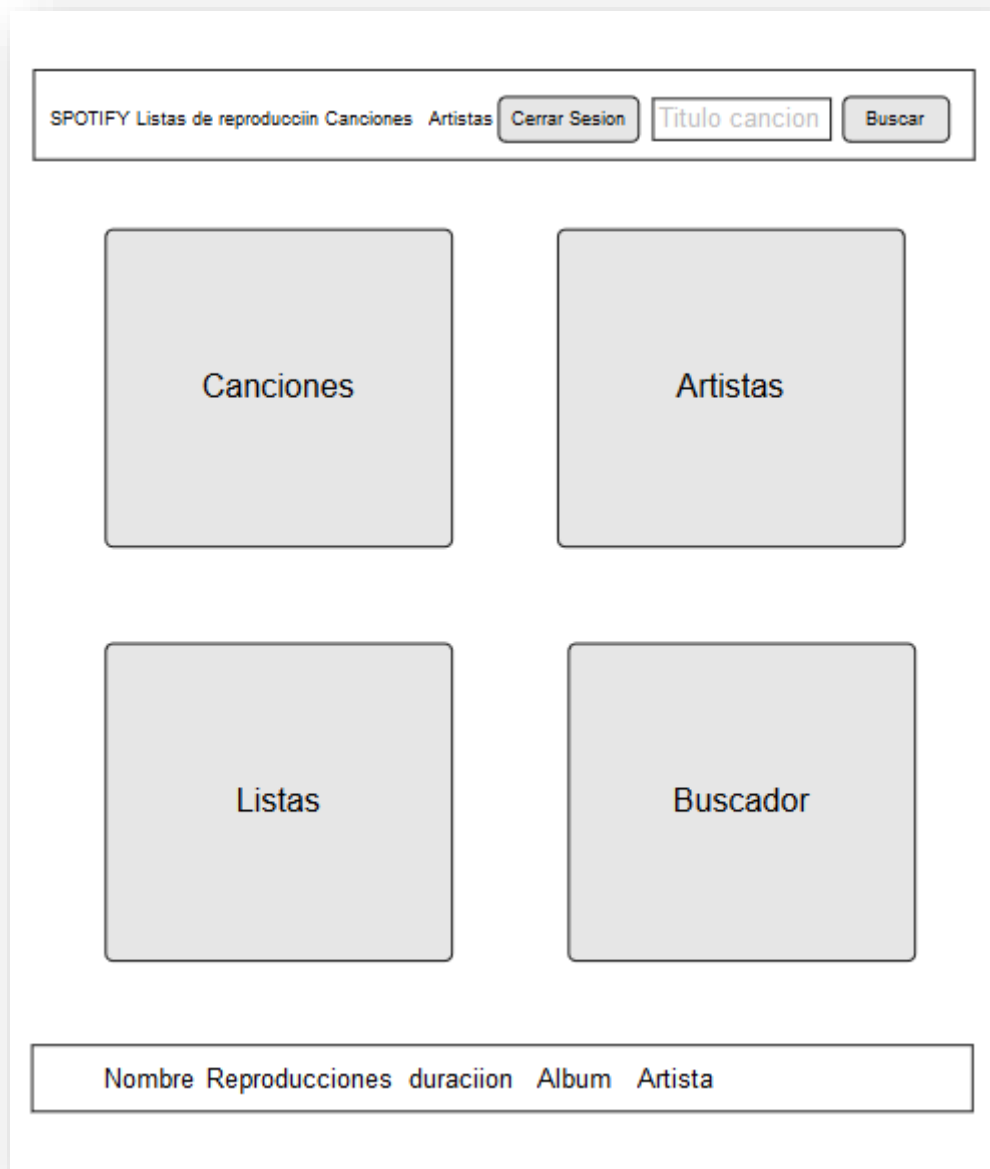
Creadores

[Inicio](#) [Registro](#) [Acerca de](#)

Nombre creador 1

Nombre creador 2

Esta página muestra la información de los creadores.



Aquí podemos ver la página principal de la parte privada del usuario. Contamos con una navegación en la parte superior para poder acceder a sus distintas opciones (Listas de reproducción, artistas, canciones y el buscador). En la parte central nos encontramos 4 botones que nos darán de manera clara acceso a las diferentes secciones y en la parte inferior encontramos la información de la canción que se está reproduciendo.

El patrón seguido para el diseño de esta página es el patrón de página principal, dado que pretendemos ofrecer al usuario acceso a todas las funcionalidades. También podemos destacar el patrón de navegación de la barra superior que nos conecta con las distintas partes del sitio.

Todas las canciones

SPOTIFY Listas de reproducciin Canciones Artistas

Cerrar Sesion

Titulo cancion

▼ id	▼ Nombre	▼ Escuchar
1	Nombre1	<input checked="" type="checkbox"/>
2	Nombre2	<input checked="" type="checkbox"/>

Nombre Reproducciones duracion Album Artista

En esta página se muestran todas las canciones de la colección, el listado esta paginado debido que la cantidad de datos que podemos obtener es demasiado alta.

El patrón principal utilizado en esta página es un patrón de listado utilizando una lista tabular o tabla(al tener muchos datos que mostrar).

Listas de Reproduccion

SPOTIFY Listas de reproducciin Canciones Artistas

Cerrar Sesion

Titulo cancion

▼ id	▼ Nombre
1	Lista reproduccion 1
2	Lista reproduccion 2

Nombre Reproducciones duraciion Album Artista

En esta página se muestran todas las listas de reproducción del usuario, el listado esta paginado debido que la cantidad de datos que podemos obtener es demasiado alta.

El patrón principal utilizado en esta página es un patrón de listado utilizando una lista tabular o tabla (al tener muchos datos que mostrar).

Musica por Artista

SPOTIFY Listas de reproducciin Canciones Artistas

Cerrar Sesion

Titulo cancion

▼ id	▼ Nombre	▼ Canciones
1	Nombre1	<input checked="" type="checkbox"/>
2	Nombre2	<input checked="" type="checkbox"/>

Nombre Reproducciones duracion Album Artista

En esta página se muestran todos los artistas, el listado esta paginado debido que la cantidad de datos que podemos obtener es demasiado alta.

El patrón principal utilizado en esta página es un patrón de listado utilizando una lista tabular o tabla (al tener muchos datos que mostrar).

Musica por artista

SPOTIFY Listas de reproducciin Canciones Artistas

Cerrar Sesion

Titulo cancion

NOMBRE ARTISTA

Descripcion artista

estilo fecha

▼ id	▼ Nombre	▼ Escuchar
1	Nombre1	<input checked="" type="checkbox"/>
2	Nombre2	<input checked="" type="checkbox"/>

Nombre Reproducciones duraciion Album Artista

En esta página se muestran todas las canciones del artista, el listado esta paginado debido que la cantidad de datos que podemos obtener es demasiado alta.

Hemos utilizado patrones de navegación y patrón de listado.

El patrón principal utilizado en esta página es un patrón de listado utilizando una lista tabular o tabla (al tener muchos datos que mostrar).

Gestion

[Usuarios](#) [Canciones](#) [Artistas](#) [Album](#)

[Cerrar Sesion](#)

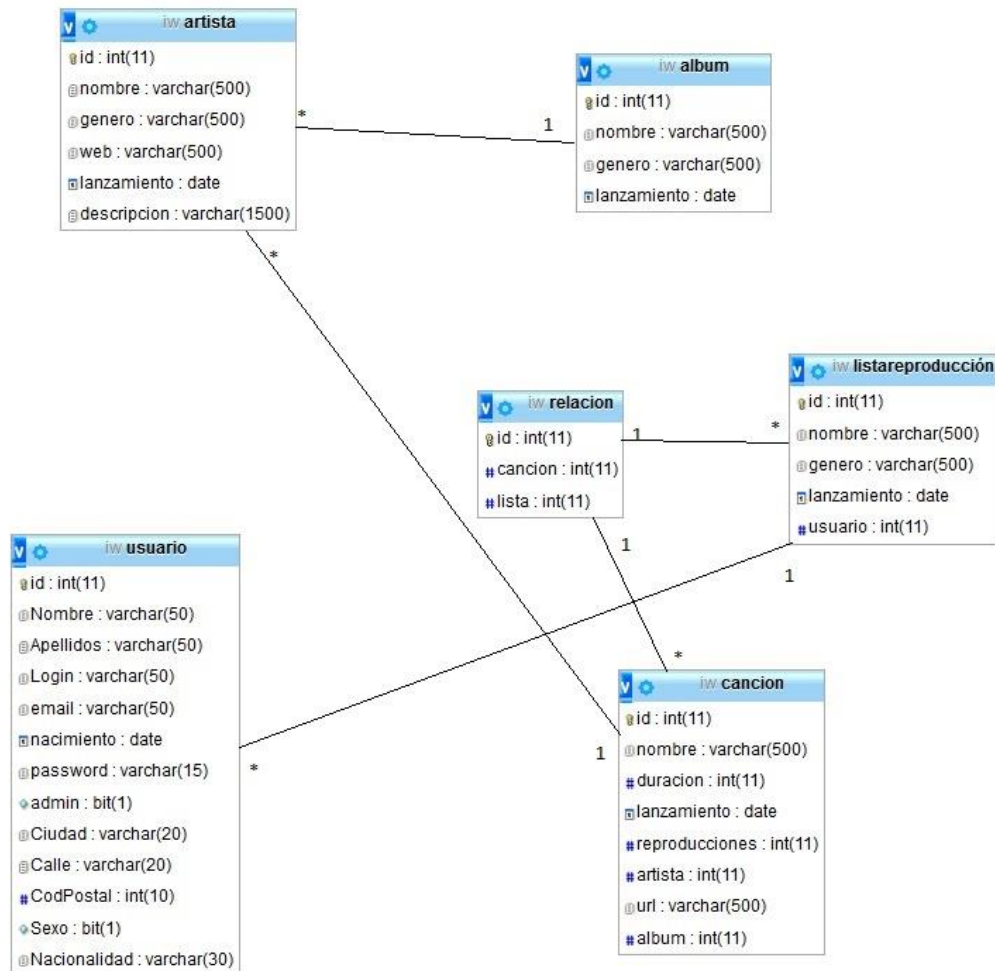
[Add](#)

▼ Campo1	▼ Campo2	▼ Campo3	▼ Acciones
1	Nombre1	valor	acciones
2	Nombre2	valor2	acciones

Por ultimo hablaremos la parte de administrador, en esta nos encontramos distintas secciones en las que podremos realizar las mismas acciones sobre ellos, por ejemplo, con usuarios podemos añadir, eliminar y editar al igual que con canciones y artistas.

Dado que estas pantallas son prácticamente iguales hemos ofrecido un mockup general.

Diagramas



Descripción técnica

Se ha usado el Framework CodeIgniter de PHP para desarrollar la aplicación ofreciéndonos facilidades de interacción con la BD y la comunicación con capas.

La arquitectura de la aplicación ha sido realizada en las siguientes capas:

Vista<->Controlador<->Modelo<->Base de Datos

- La **vista** es la encargada de renderizar la vista que el usuario utilizará para interactuar.
- El **controlador** realiza la lógica principal de la aplicación
- El **modelo** es el encargado de interactuar con la BD.

Cabe destacar que para crear una vista más intuitiva para el usuario y responsive se ha utilizado el Bootstrap CSS junto a código de CSS creado por nosotros mismos.

La base de datos se ha implementado en MySQL utilizando los clientes Workbench y phpMyAdmin.

Hablando del despliegue serían necesarios dos servidores, uno para base de datos y otro para apache donde se mantiene la aplicación web.

Por último, hemos usado Grocery CRUD para los cruds de la parte privada.

Detalles de implementación

La implementación se ha llevado a cabo utilizando un conjunto de modelo, vista, controlador para cada uno de los elementos de la aplicación. Es decir, uno para Usuario donde se controla el inicio de sesión y la parte pública; otro para la parte privada de usuario de la aplicación que hemos llamado “**inapp**” que tendría el controlador inapp pero usaría modelos independientes, uno para el buscador, otro para las canciones, otro de artistas y por último uno de listas de reproducción; el último controlador sería el de la parte privada de usuario administrador, que hace uso de un controlador y del modelo del Grocery Crud para obtener los datos.

La sesión es independiente en los dos roles de usuario que tenemos, es decir, tenemos dos tipos de usuario uno que es solo de la aplicación(básico) y otro de la aplicación y administrador. El usuario básico solo puede entrar a través del login de usuario, pero el administrador puede entrar por ambos login, pero solo se le mantiene la sesión en la parte de la aplicación por el que haya entrado.

Si un usuario intenta entrar a la aplicación sin haberse autenticado, la aplicación le manda directamente al login, y de forma contraria si ya está autenticado e intenta entrar en la parte publica le redirecciona al dashboard de la parte privada.

En ningún momento se solapan las capas para la obtención de datos ya que no lo hemos visto necesario y podría dar problemas.

Hemos usado la sesión de CodeIgniter para mantener la sesión, la canción que se escucha en cada momento y la última búsqueda en la aplicación. Las listas que no están delimitadas, como son las canciones, los artistas y las búsquedas tienen paginación realizada con las posibilidades que da CodeIgniter. Las listas que están limitadas como puede ser las canciones de un artista, o las listas de reproducción no tiene paginación ya que son listas acotadas y pensamos que de cara al usuario es mejor tener todos los datos de golpe y al ser limitadas no deberían tener un peso demasiado grande.

La aplicación respeta los estándares HTML5 y se ha programado de tal forma que se mantenga “responsive” para poder usarla desde un móvil o navegador sin problemas.

Es importante destacar que la aplicación ha sido desarrollada para que el usuario utilice el mínimo número de clicks posibles, esto es gracias a una interfaz fluida donde todas las pantallas están conectadas.

Problemas encontrados

La paginación. No conseguíamos hacer la paginación de la forma que teníamos pensada, que era siguiendo un esquema rest enviar peticiones por queryString por el navegador. La solución ha sido usar la forma predefinida de CodeIgniter, aunque haya que configurarla es mucho más sencilla de aplicar.

Reproducción de archivos desde el navegador. Actualmente se ha implementado subir los archivos al servidor pero no se ha implementado mandar estos a un reproductor.

La recuperación de contraseña no está implementada la parte de enviar un correo, ya que es necesario tener un servicio de correo para esto, pero está preparada para funcionar a falta de tener este servicio. Se comprueba la existencia del correo comprobando que existe un usuario con ese email.

Mejoras y ampliaciones

Estas son algunas de las mejoras que se podrían desarrollar para futuras ampliaciones:

- En primer lugar, la reproducción de música. Por otro lado, usar los álbumes que los hemos obviado durante la implementación para agrupar la música.
- Una sección de configuración del usuario donde se le permita editar sus parámetros y darse de baja si lo viera conveniente.
- Añadir la utilidad para el usuario de modificar sus listas de reproducción (actualmente las listas son predefinidas solo modificables por el administrador).
- La inclusión de publicidad en la interfaz de la aplicación y la personalización de la interfaz son mejoras que se podrían llevar a cabo sin mucho coste y darían un buen resultado en el conjunto de la misma.
- Posibilidad de borrarse el perfil.
- Poder ordenar las listas de reproducción.

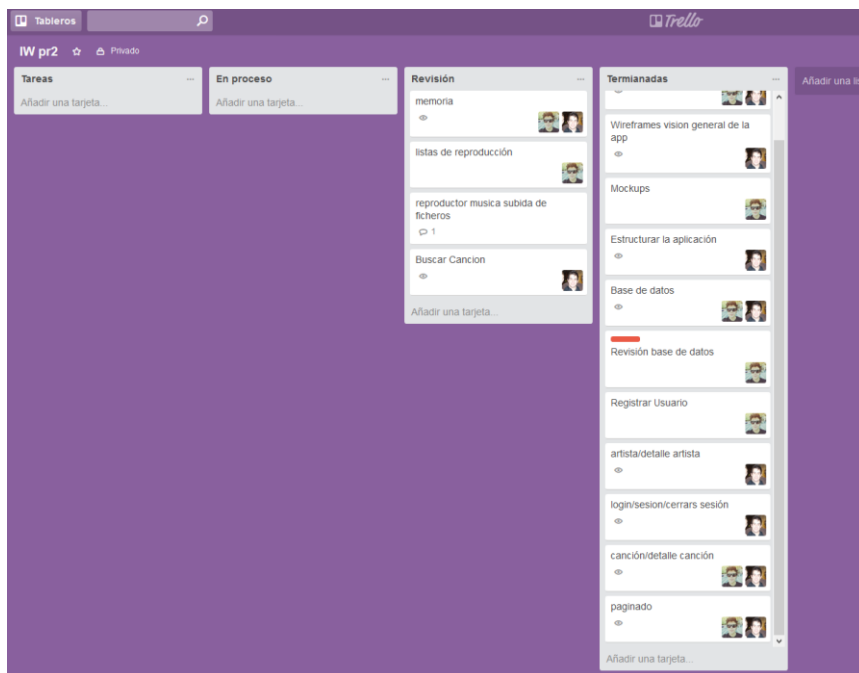
Herramientas Adicionales

Github

Hemos utilizado github para poder realizar el trabajo en equipo y tener copias seguras del trabajo.

Trello

Hemos utilizado esta herramienta para organizar el trabajo del equipo.



Referencias

https://www.codeigniter.com/user_guide/

<http://www.grocerycrud.com/documentation>

<http://getbootstrap.com/>

<http://www.w3schools.com>

<https://github.com/>

<https://trello.com/>