



[Documentación](#)

Por: Rubén Calvo Villazán

Índice

Credenciales de inicio de sesión	3
Estructura y explicación del directorio de ficheros	3
Estructura de la base de datos	6
Diseño gráfico del sitio	13
Uso de AJAX	15
Uso de cookies	16

Credenciales de inicio de sesión

IMPORTANTE: Al suministrar unas credenciales de inicio de sesión, existen usuarios registrados de la aplicación por lo tanto cuando se entre por primera vez no mostrará el mensaje de bienvenida. Para verlo dirigirse a:

<https://void.ugr.es/~rubencav1718/proyecto/database/welcome.php>

Admin:

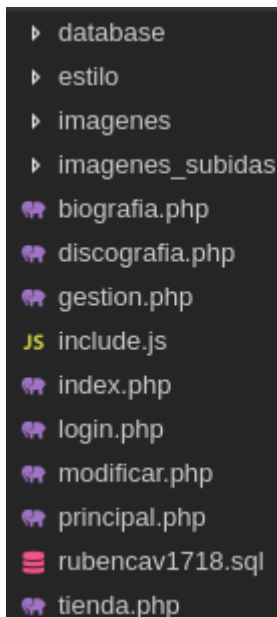
Email: prueba@gmail.com Password: 1234

Gestor:

Email: prueb1a@gmail.com Password: 1234

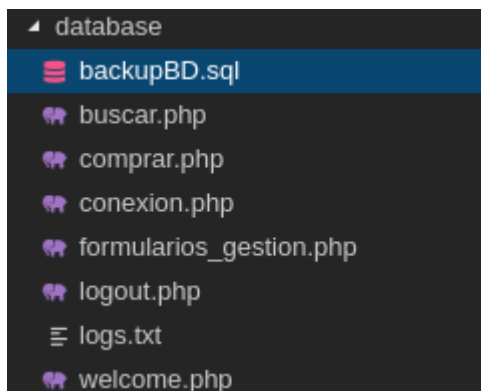
Estructura y explicación del directorio de ficheros

Encontramos la siguiente estructura:



```
▸ database
▸ estilo
▸ imagenes
▸ imagenes_subidas
🐼 biografia.php
🐼 discografia.php
🐼 gestion.php
JS include.js
🐼 index.php
🐼 login.php
🐼 modificar.php
🐼 principal.php
📄 rubencav1718.sql
🐼 tienda.php
```

‘database’ contiene los ficheros necesarios para la conexión a la base de datos y para gestiones internas del administrador y gestor, estos son ficheros que no deben ser accedidos por el usuario desde fuera.



```
▾ database
📄 backupBD.sql
🐼 buscar.php
🐼 comprar.php
🐼 conexion.php
🐼 formularios_gestion.php
🐼 logout.php
📄 logs.txt
🐼 welcome.php
```

‘welcome.php’ muestra el mensaje de bienvenida cuando la aplicación está recién instalada y no tiene usuarios.

‘buscar.php’ realiza la búsqueda en el formulario de búsqueda de [index.php](#), en este formulario se hace una petición AJAX a buscar.php (Para más información [ver uso de AJAX](#))

‘comprar.php’ muestra el formulario de compra, este fichero se accede a través de AJAX desde la [discografía](#), cuando el usuario pincha sobre el botón comprar bajo un disco, aparece en la pagina este formulario.

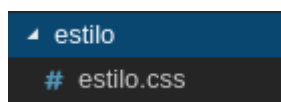
‘conexion.php’ realiza la conexión con la base de datos, también tiene la función para [guardar logs](#) ya que todos los ficheros que incluyen a este realizan alguna acción con la base de datos, luego guardamos el registro de los logs en [logs.txt](#).

‘formularios_gestion.php’ devuelve los formularios accedidos a través de AJAX desde [gestion.php](#) (para más información [ver uso de AJAX](#)).

‘logout.php’ realiza la desconexión de la aplicación del usuario que esté actualmente en uso.

‘logs.txt’ guarda los logs de la aplicación, son accedidos por [gestion.php](#)

‘estilo’ contiene el fichero .css que da estilo a toda la página, el motivo por el que se ha usado un único fichero es porque se usa AJAX en el menú para toda la página, luego el head de index.php es el mismo para todas las páginas accedidas a través del menú.



‘imagenes’ contiene imagenes de prueba para usar con la aplicación, además de las usadas por la propia aplicación.

‘imagenes_guardadas’ aqui se guardan las imagenes que han sido subidas a la aplicación, como las carátulas de discos, cartel de conciertos, artículos, etc.

‘biografia.php’ devuelve la biografía junto con la foto de cada componente del grupo a la petición hecha por AJAX desde [index.php](#) ([ver uso de AJAX](#))

‘discografia.php’ devuelve la discografía a la petición hecha con AJAX desde [index.php](#), además hace una petición a [database/comprar.php](#) en caso de que el usuario quiera comprar un disco. ([ver uso de AJAX](#))

‘gestion.php’ no imprime ningun formulario, su función es imprimir el menu para que con AJAX se imprima la opcion correspondiente, que puede ser modificar articulos, discografia, administrar usuarios, etc.

Este html se encuentra en [formularios_gestion.php](#) y según donde el usuario pinche sobre el menú, se hace unPOST con AJAX con el id del elemento del menu como variable. En

formularios_gestion.php se devuelven los formularios que generalmente son 2, añadir y modificar. El formulario de añadir se evalúa en **gestion.php** donde se hacen las inserciones, y el de modificar tiene como acción el fichero **modificar.php**, donde según las variables POST se hace lo correspondiente (parecido a **formularios_gestion.php**)
(ver uso de AJAX)

'include.js' contiene funciones de validación de formularios en JavaScript, como validar texto, validar email, etc.

'index.php' página principal, en ella se encuentra un menú que realiza peticiones por AJAX a la correspondiente selección, para ello se usa el id del elemento como url (el fichero php con contenido a obtener). También muestra los conciertos, reseñas biográficas y un **carousel hecho en JavaScript para promocionar artículos**.
(ver uso de AJAX)

'login.php' página en donde se encuentran los formularios de registrar nuevo usuario y de iniciar sesión, se hacen las comprobaciones correspondientes con JS y PHP. Si el usuario se logea en la aplicación, tiene la opción a **guardar las credenciales mediante cookies** (ver uso de cookies).

'modificar.php' contiene los formularios de modificar accedidos por **formularios_gestion.php**, esta página devuelve el formulario correspondiente según la opción elegida por el administrador o gestor en **gestion.php**

'principal.php' tiene el mismo contenido que **index.php**, se muestra cuando el usuario pincha sobre inicio en el menú de **index.php**. (ver uso de AJAX)

IMPORTANTE: **index.php** tiene este contenido porque al cargar todo el menú con AJAX, al principio aparecería la página vacía hasta que el usuario pinchase sobre el menú, de esta forma se sobrescribe el contenido.

'tienda.php' contiene el formulario de compra de la tienda, muestra todos los artículos junto con el formulario. Hace las correspondientes validaciones con JS y PHP.

Estructura de la base de datos

El modelo que se ha empleado para la base de datos es un modelo relacional. Un esquema para este modelo determina la identidad o nombre de la relación y los atributos que serán almacenados dentro de ella.

Por ello, comenzaremos a explicar las distintas tablas que encontramos en la base de datos:

Tabla
<input type="checkbox"/> admins
<input type="checkbox"/> articulos
<input type="checkbox"/> biografia
<input type="checkbox"/> canciones
<input type="checkbox"/> componentes
<input type="checkbox"/> conciertos
<input type="checkbox"/> discos
<input type="checkbox"/> gestores
<input type="checkbox"/> pedidos
<input type="checkbox"/> usuarios

Tabla 'admins'

En esta tabla se guardan todos los usuarios que son administradores, contiene los datos de cada administrador y se trabaja con ella en [login.php](#) y en [gestion.php](#) a la hora de crear o administrar usuarios

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	nombre	varchar(50)	utf8_spanish_ci		No	Ninguna
2	apellidos	varchar(100)	utf8_spanish_ci		No	Ninguna
3	telefono	int(9)			No	Ninguna
4	email 	varchar(50)	utf8_spanish_ci		No	Ninguna
5	clave_cifrada	varchar(100)	utf8_spanish_ci		No	Ninguna

nombre	apellidos	telefono	email	clave_cifrada
Nuevo	Admin	123456789	nuevo@gmail.com	MTIzNDU2
Ruben	SASD	123456789	prueba@gmail.com	MTIzNA==

Como se puede comprobar, la contraseña se guarda cifrada

Tabla ‘gestores’

Esta tabla es similar a la tabla ‘admins’ pero con los usuarios de tipo gestor, se trabaja con ella en los mismos ficheros.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	nombre	varchar(50)	utf8_spanish_ci		No	Ninguna
2	apellidos	varchar(100)	utf8_spanish_ci		No	Ninguna
3	telefono	int(9)			No	Ninguna
4	email 	varchar(50)	utf8_spanish_ci		No	Ninguna
5	clave_cifrada	varchar(100)	utf8_spanish_ci		No	Ninguna

nombre	apellidos	telefono	email	clave_cifrada
Nuevo	Admin	123123123	nuevo@gmail.com2	MTIzNA==
Armando	Casas	123456789	prueba1a@gmail.com	MTIzNA==
Batman	Superman	123456789	prueba@gmail.com	MTIzNA==

Tabla 'usuarios'

La tabla usuarios sirve para guardar la información esencial de cada usuario. Esto es el login y el tipo de usuario que es. Esta tabla se emplea para que el usuario inicie sesión en [login.php](#) usando únicamente el correo y la contraseña, extrayendo el tipo de usuario de ésta tabla (ya que el email es primary key en la tabla 'usuarios', 'admins' y 'gestores') y guardando el tipo de usuario en \$_SESSION directamente.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	email 🔑	varchar(50)	utf8_spanish_ci		No	Ninguna
2	clave_cifrada	varchar(100)	utf8_spanish_ci		No	Ninguna
3	tipo	varchar(20)	utf8_spanish_ci		No	Ninguna

email	clave_cifrada	tipo
nuevo@gmail.com	MTizNDU2	admin
nuevo@gmail.com2	MTizNA==	gestor
prueb1a@gmail.com	MTizNA==	gestor
prueb2a@gmail.com	MTizNA==	admin
prueba@gmail.com	MTizNA==	admin

Tabla 'articulos'

Contiene todos los artículos que están a la venta en la tienda.

Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
foto	varchar(100)	utf8_spanish_ci		No	Ninguna
nombre 🔑	varchar(100)	utf8_spanish_ci		No	Ninguna
precio	decimal(9,0)			No	Ninguna
tipo	varchar(50)	utf8_spanish_ci		No	Ninguna

foto	nombre	precio	tipo
imagenes_subidas/articulos/snoop1.jpg	Discos (Ed. Especial)	35	merch
imagenes_subidas/articulos/snoop3.jpg	Doggystyle (Special Ed.)	25	disco
imagenes_subidas/articulos/last-meal.jpg	Last Meal	16	disco
imagenes_subidas/articulos/snoop2.jpg	Merchandising	25	merch
imagenes_subidas/articulos/neva-left.jpg	Neva Left	11	disco

Tabla ‘discos’

Contiene todos los discos del artista, esta tabla es independiente de la de artículos luego **si en la aplicación se modifica un disco en ‘articulos’ se deberá modificar en ‘discos’ también.** Esto se hace así por si se desean subir discos exclusivos que no estén a la venta, etc.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	foto	varchar(50)	utf8_spanish_ci		No	Ninguna
2	nombre 🔑	varchar(100)	utf8_spanish_ci		No	Ninguna
3	year_s	int(4)			No	Ninguna

foto	nombre	year_s
imagenes_subidas/discos/doggystyle.jpg	Doggystyle	1996
imagenes_subidas/discos/snoop3.jpg	Doggystyle (Ed. Especial)	2018
imagenes_subidas/discos/neva-left.jpg	Neva Left	2017

Tabla ‘biografia’

Esta tabla guarda la fecha del hito histórico junto con su texto correspondiente.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
1	id 🔑	int(10)		UNSIGNED	No	Ninguna	AUTO_INCREMENT
2	fecha	varchar(20)	utf8_spanish_ci		No	Ninguna	
3	texto	varchar(10000)	utf8_spanish_ci		No	Ninguna	

id	fecha	texto
6	2017	Calvin Cordozar Broadus Jr. (Long Beach, 20 de oct...
7	1996	Snoop Dogg fue al instituto de Long Beach Polytech...

Tabla ‘canciones’

Guarda las canciones junto con el disco al que pertenecen, también su duración.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	disco	varchar(50)	utf8_spanish_ci		No	Ninguna
2	nombre 🔑	varchar(50)	utf8_spanish_ci		No	Ninguna
3	duracion	varchar(10)	utf8_spanish_ci		No	Ninguna

disco	nombre	duracion
Doggystyle	Dogg Collar	4:02
Doggystyle	Doggy Dogg World	3:36
Doggystyle	Gin And Juice	3:02
Neva Left	m.a.c.a	5:30
Doggystyle	Murder Was The Case	4:28
Neva Left	neva left	4:20
Neva Left	snoop	4:20
Doggystyle (Ed. Especial)	Still DRE	3:26

Tabla ‘componentes’

Contiene la información correspondiente a los componentes del grupo. Las [fotos de la biografía](#) se extraen de esta tabla, el apartado [Synopsis](#) de [index.php](#) muestra los nombres y reseñas biográficas de esta tabla.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	nombre 🔑	varchar(50)	utf8_spanish_ci		No	Ninguna
2	nacimiento	varchar(20)	utf8_spanish_ci		No	Ninguna
3	foto	varchar(50)	utf8_spanish_ci		No	Ninguna
4	bio_componente	varchar(500)	utf8_spanish_ci		No	Ninguna

nombre	nacimiento	foto	bio_componente
Snoop Dogg	2018-05-31	imagenes_subidas/componentes/bio.jpg	West Coast rap leyenda Snoop Dogg has come a long ...
Snoop Doggy Dogg	2018-03-09	imagenes_subidas/componentes/snoop4.jpg	Primero en el tema principal del largometraje Deep...

Tabla ‘conciertos’

Contiene la información correspondiente a los conciertos del grupo, guarda el nombre, lugar, fecha, etc. Los conciertos mostrados en [index.php](#) se extraen de esta tabla.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	cartel	varchar(50)	utf8_spanish_ci		No	Ninguna
2	fecha	varchar(20)	utf8_spanish_ci		No	Ninguna
3	hora	varchar(20)	utf8_spanish_ci		No	Ninguna
4	lugar	varchar(50)	utf8_spanish_ci		No	Ninguna
5	nombre 🔑	varchar(50)	utf8_spanish_ci		No	Ninguna
6	descripcion	varchar(500)	utf8_spanish_ci		No	Ninguna

cartel	fecha	hora	lugar	nombre	descripcion
imagenes_subidas/conciertos/concert2.jpg	2018-06-14	19:30	Londres	Granada sound	Increible concierto de Snoop Dogg en Granada sound...
imagenes_subidas/conciertos/poster.jpg	2018-08-18	17:30	Madrid	Madrid Arena	Impresionante escenario planeado para el concierto...
imagenes_subidas/conciertos/concert1.jpg	2018-12-13	22:22	California	Tomorrowland	Concierto de Snoop Dogg en California para el fes...

Tabla 'pedidos'

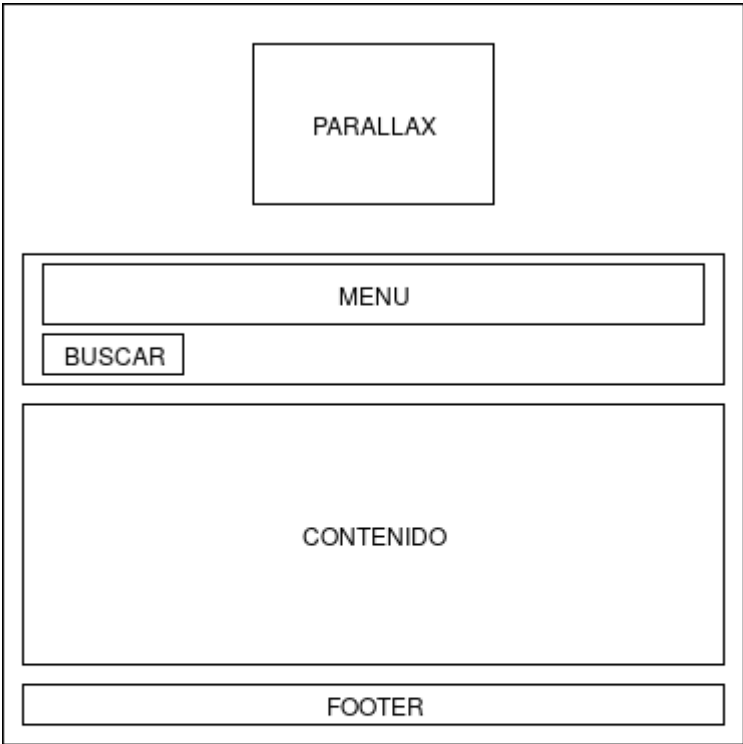
Esta tabla guarda los pedidos realizados, los datos del gestor y el estado del pedido. El estado puede ser '**pendiente**', '**aceptado**' o '**denegado**' y se trabaja con esta tabla en [tienda.php](#) cuando se realiza una compra, y en [gestion.php](#) en el lado del gestor cuando administra los pedidos pendientes.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
1	correo_cliente	varchar(50)	utf8_spanish_ci		Sí	NULL	
2	fecha_pedido	date			Sí	NULL	
3	fecha_gest	date			Sí	NULL	
4	nombre_gest	varchar(50)	utf8_spanish_ci		Sí	NULL	
5	apellidos_gest	varchar(50)	utf8_spanish_ci		Sí	NULL	
6	estado	varchar(20)	utf8_spanish_ci		Sí	NULL	
7	email_enviado	varchar(1000)	utf8_spanish_ci		Sí	NULL	
8	nombre_producto	varchar(50)	utf8_spanish_ci		Sí	NULL	
9	id 🔑	int(11)			No	Ninguna	AUTO_INCREMENT

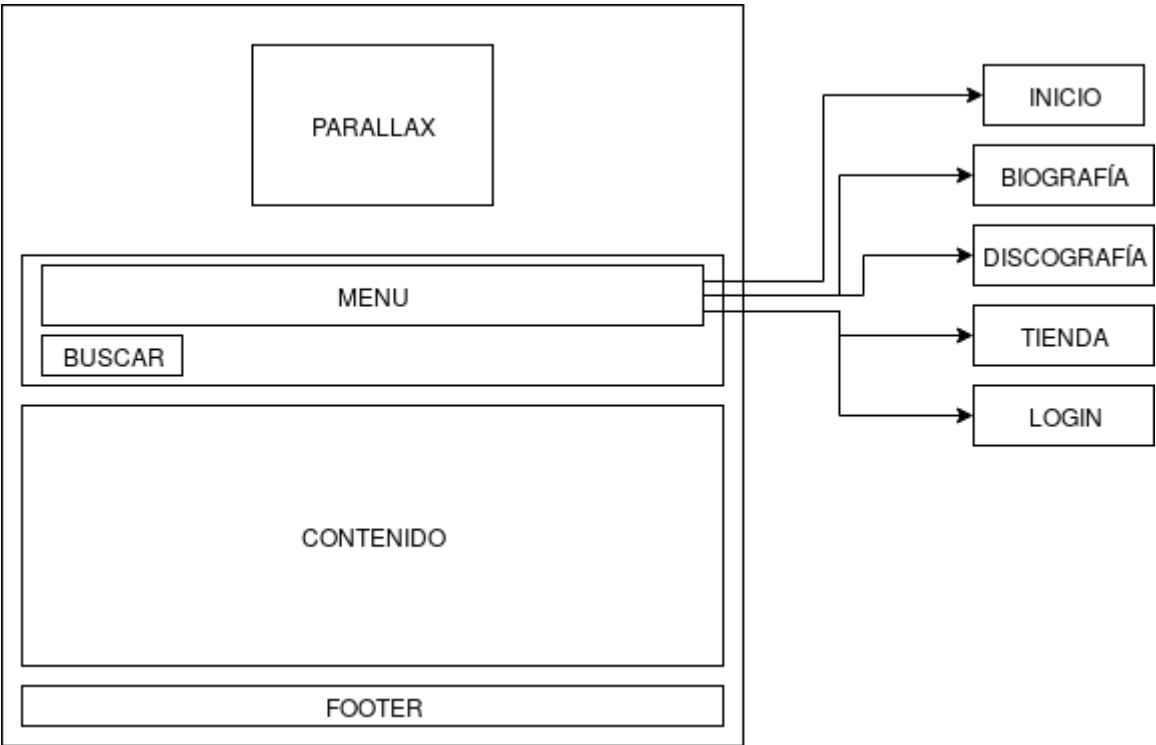
correo_cliente	fecha_pedido	fecha_gest	nombre_gest	apellidos_gest	estado	email_enviado	nombre_producto	id
prueba@gmail.com	2018-06-08	2008-06-18	Armando	Casas	aceptado	Sisisi espero que te guste	Doggystyle	3
prueba@gmail.com	2018-06-08	2008-06-18	Armando	Casas	denegado	Lo siento pero no quedan existencias	Doggystyle	4
prueba@gmail.com	2018-06-08	2008-06-18	Armando	Casas	denegado	Sin existencias	NevaLeft	5
snoop@gmail.com	2018-06-10	2010-06-18	Armando	Casas	aceptado	Enhorabuena, le enviaremos su pedido	Discos (Ed. Especial) Last Meal Merchandis	6

Diseño gráfico del sitio

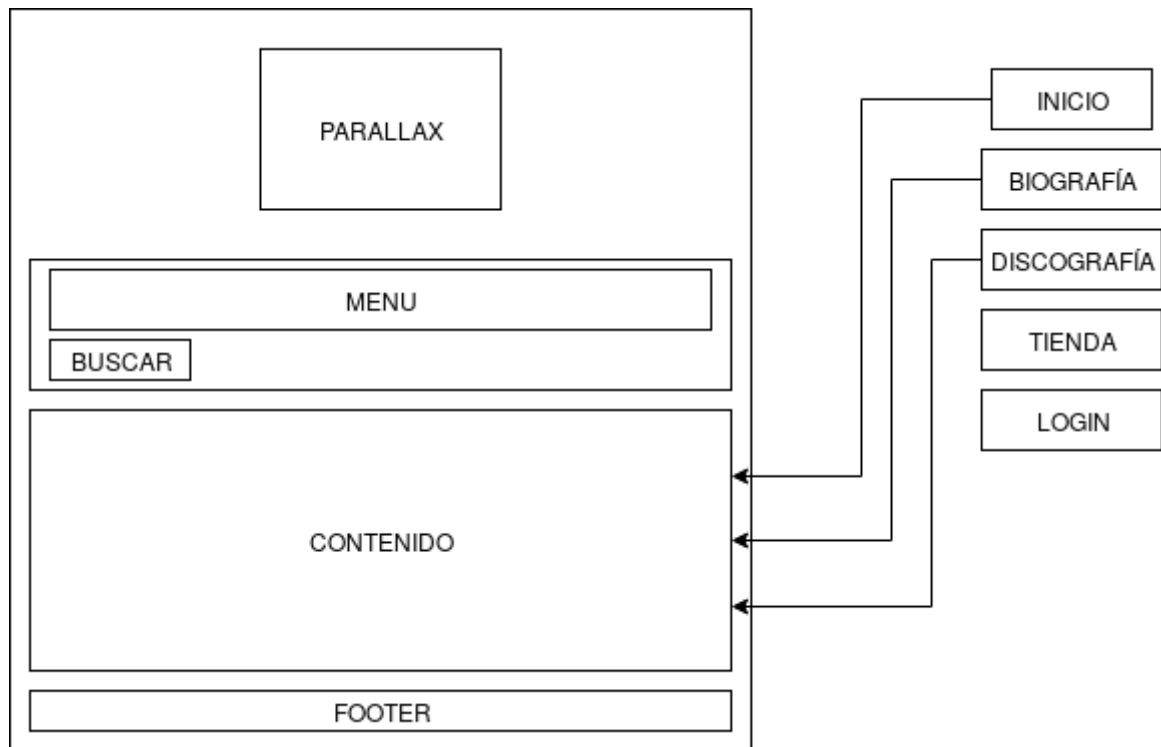
Diseño general del sitio (index.php)



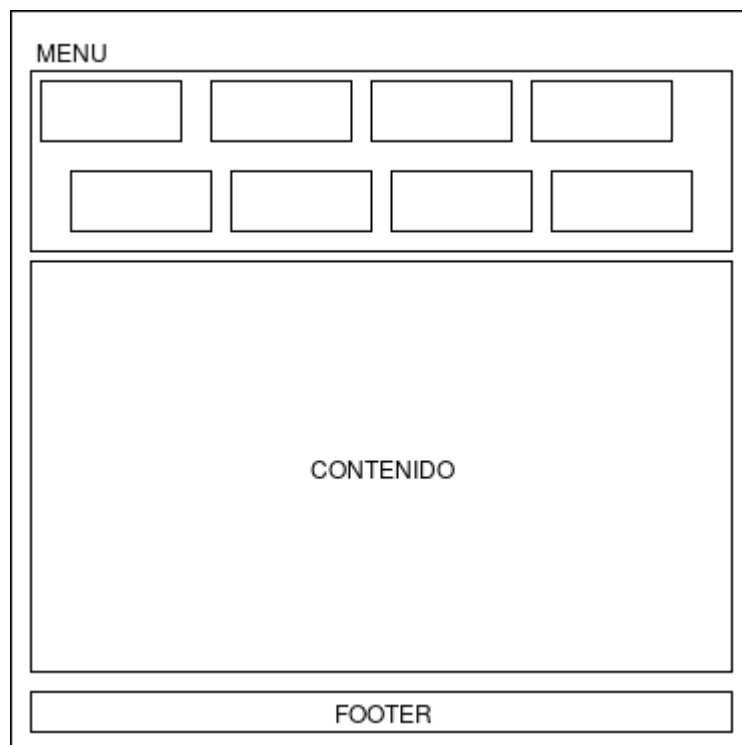
Páginas accedidas por el menú



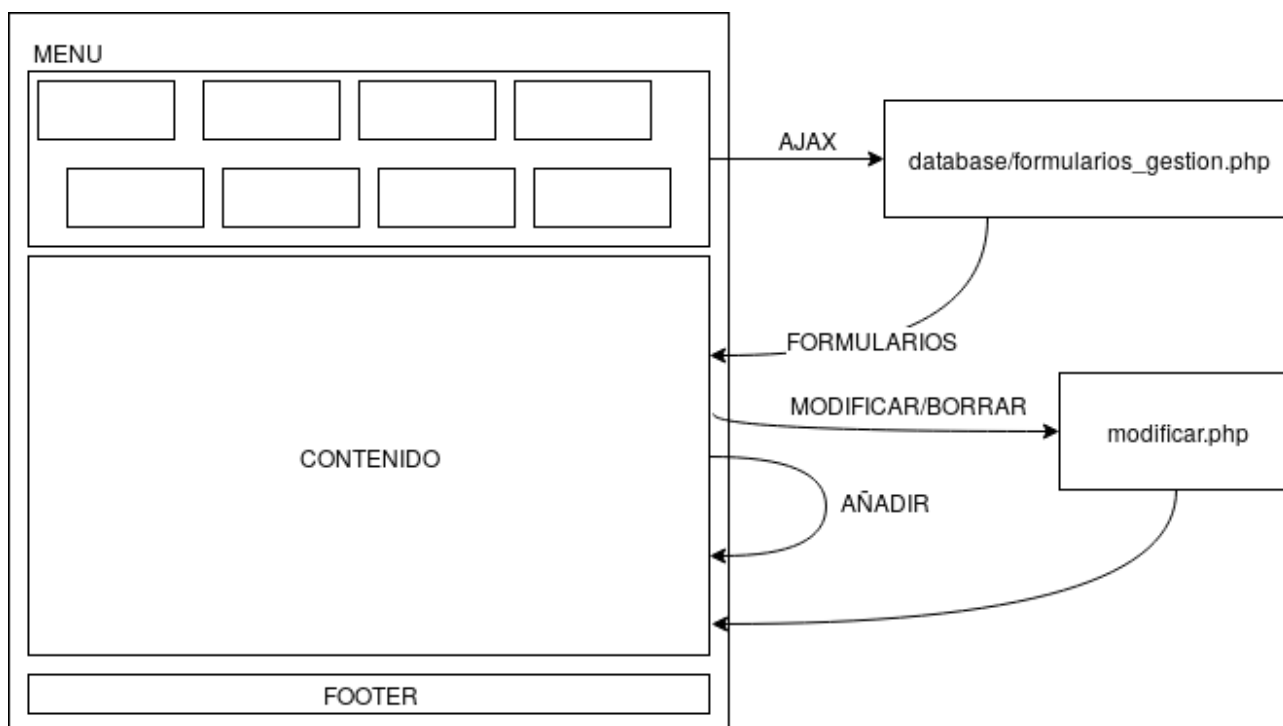
Páginas que se cargan en el contenido a través de AJAX



Página de administración (gestion.php)



AJAX en la página de administración (gestion.php)



Uso de AJAX

Se utiliza AJAX en los siguientes sitios:

- En el menú de [index.php](#), según donde pinche el usuario se extrae el id de ese elemento, ese id es el nombre de un fichero php que se usará como url en la petición de AJAX. Se obtiene el contenido de dicho fichero y se muestra en el section con id="imprimirAqui"
- Botón de compra de disco, esta es una petición AJAX **encadenada a la anterior**, si tras hacer la petición anterior nos encontramos en [discografía](#), existe un botón de compra tal que si el usuario pincha sobre él, se muestra el formulario de compra. Este botón tiene una clase con el nombre del disco (puesto dinámicamente con PHP) y hace una petición a [database/comprar.php](#). Como variable POST a enviar, se usa el atributo clase que es el nombre del disco sin espacios (Si hay espacios se toma como dos clases), y se hace el correspondiente envío al fichero de compras ([comprar.php](#)). Este formulario de compra se muestra debajo de cada disco en un section que tiene como clase 'imprimir' + nombre del disco.
- En el formulario de búsqueda aquí se **encadena AJAX hasta tres veces**. En el menú de [index.php](#) mostramos un botón de buscar, si el usuario pincha sobre este botón, lo ocultamos y hacemos un envío AJAX a [buscar.php](#) para que nos devuelva un input de texto y un menú desplegable para elegir entre disco o concierto. Si el usuario pincha sobre el menú desplegable, se hace un nuevo envío AJAX al fichero [buscar.php](#)

devolviéndonos campos más específicos para la búsqueda (fecha o lugar en caso del concierto) y un nuevo botón para realizar la búsqueda. Si el usuario pincha sobre el botón, se encadena una tercera petición AJAX a [buscar.php](#) con los parámetros a buscar y nos devuelve el resultado de la búsqueda, que se imprime en el tercer section consecutivo.

Según los parámetros introducidos, se realizará una búsqueda u otra (recomendable ver el fichero [buscar.php](#))

- En el menú de administrador y gestor se usa AJAX para mostrar los formularios correspondientes a la operación que se desea realizar. Desde el fichero [gestion.php](#) se envía la petición AJAX extrayendo el campo id de los elementos del menú. Esta petición se realiza al fichero [formularios_gestion.php](#) y según el valor del id, se realiza con esa variable POST. El fichero [formularios_gestion.php](#) comprueba ese valor y devuelve los formularios o datos correspondientes a esa acción.
IMPORTANTE: Las acciones sobre añadir se realizan en [gestion.php](#), las acciones sobre modificar o eliminar elemento (u otras) se realizan en [modificar.php](#) (ver [diagrama del diseño](#))

Uso de cookies

Se utilizan cookies en [login.php](#) guardadas con JavaScript. Si el usuario acepta el ‘confirm’ que se ejecuta cuando el usuario pincha sobre iniciar sesión, se crea una cookie con el email y otra con la contraseña. Ambos se codifican con `encodeURIComponent` y se decodifican con `decodeURIComponent`. Al acceder a [login.php](#) se comprueba si existen éstas cookies, si es así, se muestran en los campos del formulario de login.