



DECSAI

Departamento de Ciencias de la Computación e I.A.

Universidad de Granada

Programación Web

Grado en Ingeniería Informática

Práctica Evaluable II



Nombre: Rubén

Apellidos: Jiménez Ortega

Grupo: 2

DNI: 53598079-Y

Correo: rubenjo7@correo.ugr.es

Practica Evaluable 2:

Descripción del problema:

Se nos pide la adaptación de la práctica anterior para la realización de una tienda de música digital.

Para ello debemos de poner en práctica los conocimientos de PHP y JavaScript así como las conexiones con las bases de datos. Se nos piden unos cuantos puntos en esta práctica:

1. A partir del formulario se hizo en la primera práctica evaluables para que los clientes de la tienda digital se den de alta en la web, darlos de alta en la base de datos de clientes.
2. Modificar la página principal para permitir identificarse al cliente y abrir así una sesión, manteniéndola activa hasta que el usuario, de alguna forma, cierre la misma.
3. Crear un formulario para que cualquier usuario identificado pueda incorporar comentarios a los discos y que éstos se muestren al acceder el detalle del mismo. En cada comentario aparecerá la hora y el día, el nombre de usuario, y el propio texto del usuario.
4. Validar la corrección de todos los formularios (con JavaScript). En relación con los formularios, deben existir campos obligatorios, los comentarios a los discos deben tener una longitud máxima, deben existir valores numéricos, números de teléfonos y direcciones de correo electrónico o URLs (todos validados según su formato correspondiente).
5. Debe existir el usuario admin, que se encargará de gestionar los discos en venta. Así, mediante un nuevo formulario podrá dar de alta nuevos discos que se asignarán a las secciones correspondientes. Por razones de simplicidad, sólo se podrán asignar a secciones Rock y Pop.
6. Al pasar el ratón por la denominación de un disco (en la pagina donde se da toda la información del disco), aparecerá una ventana emergente con las canciones que contiene.

Descripción de la solución:

Modularización de código:

Lo primero que pasare a hacer será modularizar el contenido de la anterior página web. De manera que en el caso de que quiera cambiar algo de la cabecera o el pie de página solo lo tenga que cambiar en el mismo sitio.

Llevo a cabo la separación de distintos elementos, por un lado está el clases/creador.php, para hacer los includes de todos los documentos javascript, conexiones a base de datos a clases, cambiar el título, incrustar estilos , creación de sesiones y todo aquello que se necesite:

```
<?php
class Creador{
    private $titulopagina;
    /*Constructor por defecto*/
    function __construct(){
        $this->titulopagina = 'Tienda Música';
    }
    /*Destructor*/
    function __destruct() {
    }
    /*Set del titulo*/
    function set_titulo($titulo){
        $this->titulopagina = $titulo;
    }
    /*Funcion creadora*/
    function crear(){
        session_start();
        echo'<!DOCTYPE html>
        <html lang="es">
        <head>
        <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
        <title>'.$this->titulopagina.'</title>
        <link href="style.css" rel="stylesheet" type="text/css" />
```

```

        <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.4.2/jquery.min.js?
ver=1.4.2"></script>
        <script src="js/login.js"></script>
        <script type="text/javascript" src="scripts.js"></script>
    </head>';
    require_once("conexionpdo.php");
    require_once("cdisco.php");
    require_once("cusuario.php");
    require_once("ccanciones.php");
}
}
?>

```

En esta práctica he usado como podemos ver jquery enlazado directamente a la página web de google, con la versión 1.4.2, simplemente lo he usado para que el login sea flotante, ya lo explicare más adelante. Podemos observar la conexión PDO con la base de datos, aunque esta es más lenta que la Mysql. Y una clase que he creado para el control de los discos, usuarios y canciones, ya que repetía muchas acciones a lo largo de la web.

En la página web podemos observar como tiene una cabecera, el nombre de la tienda, el login y las secciones de la tienda.

Y un pie de página con las descripciones que se nos pedía en la primera práctica.

Creación de la base de datos:

Para la creación de la base de datos he utilizado las siguientes tablas:

```

CREATE TABLE `canciones` (
  `idCanciones` int(11) NOT NULL,
  `nombre` varchar(100) NOT NULL,
  `idDisco` int(6) NOT NULL
);

```

```

CREATE TABLE `comentarios` (
  `idComentario` int(6) NOT NULL,
  `idDisco` int(6) NOT NULL,
  `idUsuario` int(6) NOT NULL,
  `comentario` varchar(140) NOT NULL,
  `fecha` date NOT NULL,

```

```

`hora` time NOT NULL
);

CREATE TABLE `disco` (
  `idDisco` int(11) NOT NULL,
  `titulo` varchar(120) NOT NULL,
  `precio` varchar(50) NOT NULL,
  `fecha` date NOT NULL,
  `tipo` int(4) NOT NULL,
  `pos` int(2) NOT NULL,
  `imagen_url` varchar(100) NOT NULL,
  `artista` varchar(20) NOT NULL
);

CREATE TABLE `usuarios` (
  `idUsuario` int(6) NOT NULL,
  `nombreUsuario` varchar(40) NOT NULL,
  `contrasena` varchar(40) NOT NULL,
  `email` varchar(40) NOT NULL,
  `nombre` varchar(40) NOT NULL,
  `apellidos` varchar(40) NOT NULL,
  `telefono` varchar(20) NOT NULL,
  `tipo` int(2) NOT NULL
)

```

La tabla de comentarios lleva el comentario en sí, que cliente los escribió, la fecha y hora del mismo y un identificador del disco..

La tabla usuarios guarda los datos de los usuarios así como sus datos, decir que todas las contraseñas las tengo encriptadas para una mayor seguridad.

Disco, es donde se guardan los productos, con su título, precio y artista, además de la zona en la que se posiciona en la página de cada sección 1 si a la izquierda, 2 si en la derecha, y la sección a la que pertenece mediante un número, esto podría haberse mejorado con la incorporación de otra tabla con las secciones y algunas cosas más, pero lo dejo más sencillo.

Y por último la tabla de canciones, que relaciona las canciones que hay con el disco que las incluyen.

Formulario de registro para usuarios:



Formulario de registro para compradores:

Usuario: *

Contraseña: *

Nombre: *

Apellidos: *

E-mail: *

Telefono: *(Ejemplo 958112233)

*Campos obligatorios

He realizado el formulario de registro de nuevos usuarios, en el login que posteriormente explicare podemos hacer click en una nueva cuenta, y nos saldrá el formulario de la imagen, este controla todos los campos mediante Javascript controlado que estén rellenos, que el email sea de la forma correcta y el número de teléfono sea numérico.

Una vez se rellenen los datos correctamente el sistema mediante una conexión php comprueba que usuario no existe ya en la base de datos, si existe avisa y devuelve el formulario de nuevo.

En caso de que no exista el usuario, inserta en la tabla de usuarios el nuevo usuario.

Login de acceso a compradores:

Para el acceso de usuarios a la página web, he creado un login jquery que he adaptado de internet, decir que primeramente puse que los accesos de los login simplemente llevaran required (HTML5) en vez de JavaScript porque no necesitamos para dos campos vacíos controlarlos de esta manera, es más sencillo y ágil en este caso usar el required.

Como podemos observar el login está de esta manera en la cabecera, y cuando hacemos click sobre él nos aparece de la siguiente forma:



Login

Usuario

Contraseña

[Crear una nueva cuenta](#)

Cuando insertamos los datos usu/usu vemos que hace el login de la siguiente manera.



Pulsando el botón de cerrar sesión, se cierra la sesión actual y debemos de logearnos de nuevo.

Además de todo esto, el usuario dispone de una zona propia, en la cual puede consultar sus datos, editarlos o incluso darse de baja:

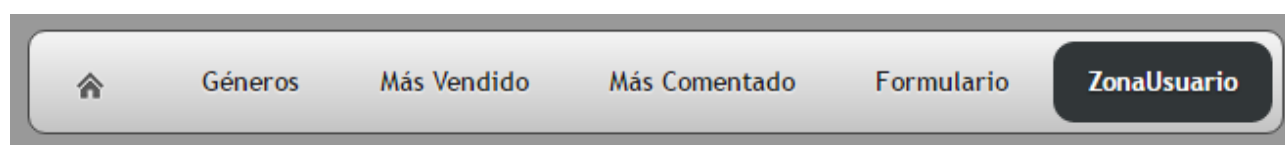




Tabla de Usuarios

NOMBRE DE USUARIO	EMAIL	NOMBRE	APELLIDOS	TELEFONO	EDITAR	BORRAR
usu	usuario@normal.es	Usuario	Normal	989898891		

Editando el Usuario

Contraseña (50 caracteres maximos):	<input type="text"/>
Nombre (50 caracteres maximos):	<input type="text" value="Usuario"/>
Apellidos (50 caracteres maximos):	<input type="text" value="Normal"/>
Email (50 caracteres maximos):	<input type="text" value="usuario@normal.es"/>
Telefono (50 caracteres maximos):	<input type="text" value="989898891"/>
<input type="button" value="Enviar usuario"/>	

El usuario ha sido borrado con éxito.

Comentarios:

Para el uso de comentarios he usado un formulario que aparece cuando el comprador esta logeado, en caso de no estarlo muestra un aviso:

opiniones:

- Usuario: usu Dia: 2016-06-04 Hora 22:17:25:
- Comentario: fdsfsdfdsfs
- Usuario: usu Dia: 2016-06-04 Hora 20:58:45:
- Comentario: Genial!
- Usuario: usu Dia: 2016-06-04 Hora 17:07:19:
- Comentario: sadasd

Para escribir comentarios debes de tener una cuenta

Y cuando el usuario se logea le permite insertar el comentario:

opiniones:

- Usuario: usu Dia: 2016-06-04 Hora 22:17:25:
- Comentario: fdsfsdfdsfs
- Usuario: usu Dia: 2016-06-04 Hora 20:58:45:
- Comentario: Genial!
- Usuario: usu Dia: 2016-06-04 Hora 17:07:19:
- Comentario: sadasd

Escribe un comentario

Limitado a 140 caracteres, no podra escribir mas

Enviar

Cuando se inserta el comentario muestra un aviso de que el comentario ha sido insertado y ya lo podemos ver como se ha insertado.

Por último añadir, que también tengo una función que cuenta los comentarios y los muestra en el icono de comentarios:



Validación de formularios:

La validación de los distintos formularios lo hago con Javascript. La máxima anchura de caracteres lo controlo mediante un limitador que no te deja insertar mas de X longitud.

```
function limita(area_texto,max)
{
    if(area_texto.value.length>=max)
    {area_texto.value=area_texto.value.substring(0,max);
    }
}
```

Esta función lo que hace es cortar los caracteres introducidos cada vez que introduces alguno si la longitud total de caracteres es mayor que los puestos por tope lo corta.

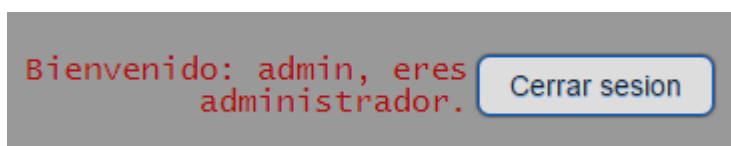
Para la validación del registro de usuarios he hecho una validación y muestro los errores abajo sin los alert.

- function validacion() → Valida el formulario de nuevos usuarios.
- function validarComentario(formulario) → Valida que el usuario registrado realice un comentario de manera correcta.
- function validacionDiscos() → Valida que el administrador inserte un disco de manera correcta.
- function validacionDiscosEditados() → Valida que el administrador edite un disco de manera adecuada.
- function validacionUsuario() → Valida que el usuario pueda editar sus datos personales de manera correcta.

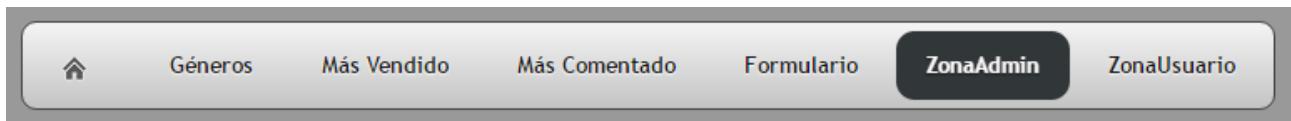
Para la validación de los comentarios, de los discos o edición, ya sea usuario o administrador, he usado la misma técnica que evalúe el formulario y que si retorna falso por algún error este no se envía. Pero en la inserción de los comentarios, inserción de discos, editado de discos o de usuario esto se avisa mediante alert y hace el foco sobre el primer campo con errores.

Usuario admin:

Para el usuario admin he creado un apartado. El usuario y la contraseña por defecto es admin / admin, una vez logueado nos aparece lo siguiente:



Y en la barra de navegación además de la zona de usuario que también aparecía para un usuario normal, nos aparece lo siguiente:



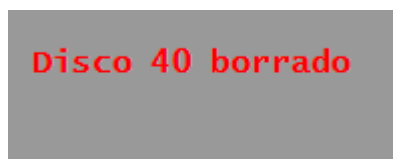
Al acceder podemos ver el panel de control completo, todo se hace en un mismo php, en caso de que no este logeado como administrador no podrá hacer nada aunque conozca todas las id y todas las variable get, sin el usuario y la contraseña no se podrá acceder.

Insertar nuevo disco

Tabla de disco

ID DISCO	TITULO	PRECIO	SECCION	COLUMNA EN SECCION	ARTISTA	CANCIONES	EDITAR	BORRAR
36	Rumba a lo desconocido	14	Pop	Izquierda	Estopa	0		
37	No lo se	20	Rock	Izquierda	Cage the Elephant	0		
38	s	5	Clasica	Izquierda	as	0		
39	a	9	Metal	Derecha	a	0		
40	sada	12	Pop	Izquierda	das	0		
87	a	12	Pop	Derecha	a	5		
92	dsa	12	Pop	Izquierda	a	5		
117	a	15	Pop	Izquierda	a	1		

Para borrar un disco simplemente damos al cubo de la basura con una X, este borrara y avisara de que ha borrado el producto.



Para editar simplemente se da al papel con un folio y el producto aparecerá relleno para editarlo.

Editando el Disco	
Titulo (120 caracteres maximos):	Rumba a lo desconocido
Precio (50 caracteres maximos):	14
Artista (50 caracteres maximos):	Estopa
Seccion de entrada:	Pop ▼
Columna en principal:	Izquierda ▼
URL completa imagen (100 caracteres maximos):	Discos/disco2.jpg
<input type="button" value="Enviar disco"/>	

Para insertar disco debe pulsar en el icono con el símbolo + y saldrá un formulario similar que para editar pero vacío completamente, esta insertara el nuevo disco junto con sus canciones. Como mínimo habrá que introducir una canción.

Titulo (120 caracteres maximos):																					
Precio (50 caracteres maximos):																					
Artista (50 caracteres maximos):																					
Seccion de entrada:	Pop ▼																				
Columna en principal:	Izquierda ▼																				
URL completa imagen (100 caracteres maximos):																					
Canciones (100 caracteres maximos):	<table><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr></table>																				
<input type="button" value="Enviar disco"/>																					

Tanto el insertado, borrado y editado tienen sus mensajes de aviso.

Ventana emergente con canciones:

Las canciones que aparecen en estas ventanas son las que se insertan junto con el disco, es decir, que tienen el mismo idDisco:



Como se podrá apreciar mejor en la web, vemos que estas canciones son las mismas que aparecen al lado de la imagen del disco:



//Funcion de ventana emergente

function showdiv(event,text) → hay que pasarle un evento y un texto con el contenido que queremos que nos muestre.

```
{
```

```
//determina un margen de pixels del div al raton  
margin=5;
```

```
//La variable IE determina si estamos utilizando IE  
var IE = document.all?true:false;
```

```
var tempX = 0;
```

```

var tempY = 0;

//document.body.clientHeight = devuelve la altura del body
if(IE)
{ //para IE
    IE6=navigator.userAgent.toLowerCase().indexOf('msie 6');
    IE7=navigator.userAgent.toLowerCase().indexOf('msie 7');
    //event.y|event.clientY = devuelve la posicion en relacion a la parte superior visible del
navegador
    //event.screenY = devuelve la posicion del cursor en relaciona la parte superior de la pantalla
    //event.offsetY = devuelve la posicion del mouse en relacion a la posicion superior de la caja
donde se ha pulsado

    if(IE6>0 || IE7>0)
    {
        tempX = event.x
        tempY = event.y
        if(window.pageYOffset){
            tempY=(tempY+window.pageYOffset);
            tempX=(tempX+window.pageXOffset);
        }else{

tempY=(tempY+Math.max(document.body.scrollTop,document.documentElement.scrollTop));

tempX=(tempX+Math.max(document.body.scrollLeft,document.documentElement.scrollLeft));
        }
    }else{
        //IE8
        tempX = event.x
        tempY = event.y
    }
}else{ //para netscape
    //window.pageYOffset = devuelve el tamaño en pixels de la parte superior no visible (scroll) de
la pagina
    document.captureEvents(Event.MOUSEMOVE);
    tempX = event.pageX;

```

```

    tempY = event.pageY;
}

if (tempX < 0){tempX = 0;}
if (tempY < 0){tempY = 0;}

// Modificamos el contenido de la capa
document.getElementById('flotante').innerHTML=text;

// Posicionamos la capa flotante
document.getElementById('flotante').style.top = (tempY+margin)+"px";
document.getElementById('flotante').style.left = (tempX+margin)+"px";
document.getElementById('flotante').style.display='block';
return;
}

/**
 * Funcion para esconder el div
 */
function hiddenDiv()
{
    document.getElementById('flotante').style.display='none';
}

```

Publicidad mediante JAVASCRIPT:

He creado un archivo en JavaScript para que la publicidad varie cada 4s.

```

var i = 0;
var ruta_slider1 = new Array();

ruta_slider1[0] = "images/banner_lateral.jpg";
ruta_slider1[1] = "images/banner_lateral2.jpg";
ruta_slider1[2] = "images/banner_lateral3.jpg";

function cambiarImagen()

```

```

{
    document.slide.src = ruta_slider1[i];

    if (i < ruta_slider1.length - 1)
        i++;
    else
        i = 0;
    setTimeout("cambiarImagen()", 4000);
}
window.onload = cambiarImagen;
Slider.js

```

Este scrip es llamado desde los archivos principales para que se ejecute en la pagina principal, las diferentes secciones y en cada disco.

Play, pause, mute y on:

Este scrip es llamado en cada disco, para que se reproduzca la canción que queramos. Básicamente son dos botones de dos posiciones:



Con estas imágenes ya podemos imaginarnos que es lo que hacen. El script que he usado es el siguiente:

```

<script>
var audio, playbtn, mutebtn;
function initAudioPlayer(){
    audio = new Audio();
    audio.src = "musica/Pastillas_para_dormir.mp3";
    playbtn = document.getElementById("playpausebtn");
    mutebtn = document.getElementById("mutebtn");
    playbtn.addEventListener("click", playPause);
    mutebtn.addEventListener("click", mute);

    function playPause(){
        if(audio.paused){
            audio.play();
            playbtn.style.background = "url(images/pause.png) no-repeat";
        }else{
            audio.pause();
            playbtn.style.background = "url(images/play.png) no-repeat";
        }
    }

    function mute(){

```

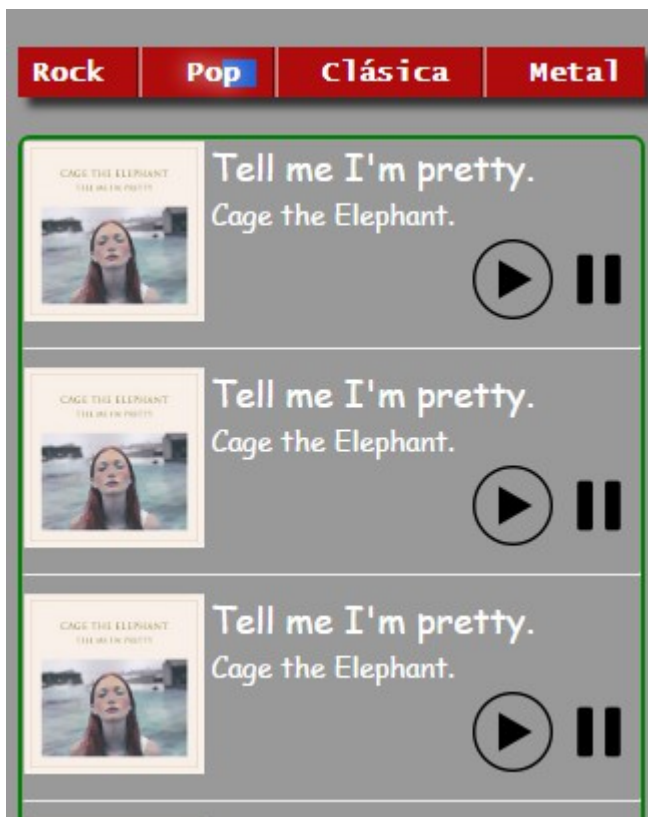
```

        if(audio.muted){
            audio.muted = false;
            mutebtn.style.background = "url(images/on.png) no-repeat";
        }else{
            audio.muted = true;
            mutebtn.style.background = "url(images/mute.png) no-repeat";
        }
    }
}
window.addEventListener("load", initAudioPlayer);

</script>

```

Cambio de submenú con JQUERY:



La función que he usado ha sido la siguiente:

```

<script src="http://code.jquery.com/jquery-2.1.4.js"></script>
<script type="text/javascript">
jQuery(document).ready(function() {
    $(".oculto2").hide();
    $(".inf2").click(function() {
        var nodo2 = $(this).attr("href");

        if ($(nodo2).is(":visible")) {
            return false;
        }else{
            $(".oculto2").hide("slow");
            $(".visible2").hide("slow");
            $(nodo2).fadeToggle("fast");
        }
    });
});

```



```
        return false;
    }

    if ($(nodo2).is(":visible")){
        return false;
    }else{
        $(".visible2").hide("slow");
        $(nodo2).fadeToggle("fast");
        return false;
    }
});

});
</script>
```

Conclusiones:

PHP es un lenguaje de programación muy potente y con una versatilidad increíble, además de existir muchísimas funciones y librerías, además hoy en día es el lenguaje más utilizado de programación en el servidor del mundo, tiene bastante facilidad pero él no ser ordenado puede traerte más de un problema.