



Aplicaciones Web

pachangafutbol.esy.es

Andrés Herranz González

Axel Junestrand Leal

Borja Lorenzo Fernández

Guillermo Rius García



Índice

1	Introducción.....	3
2	Pachanga.....	4
2.1	Página principal.....	4
2.1.1	Registrarse.....	5
2.1.2	Iniciar Sesión.....	6
2.2	Página de inicio.....	6
2.2.1	Diseño común.....	7
2.2.2	Notificaciones.....	7
2.2.3	Barra de búsqueda.....	8
2.2.4	Lista de partidos.....	8
2.3	Mi perfil.....	9
2.3.1	Puntuar un partido.....	10
2.4	Crear un partido.....	10
2.4.1	Detalles de un partido.....	11
2.5	Mis partidos.....	13
2.6	Mejores jugadores.....	14
3	Arquitectura de la aplicación.....	15
3.1	Diseño.....	15
3.1.1	Ejemplo de uso (MVC).....	17
3.2	Base de datos.....	19
3.2	Base de datos.....	19
3.3	Decisiones de diseño.....	20
3.3.1	Diseño responsive.....	21
3.4	Otros apartados.....	22
3.4.1	Seguridad.....	22
3.4.2	AJAX y JSON.....	24
3.4.3	Validadores y accesibilidad.....	25
3.4.4	Search Engine Optimization (SEO).....	28
4	Instrucciones de uso.....	28
4.1	Site Map.....	29
5	Conclusión.....	30



1 Introducción

Pachanga es un proyecto desarrollado para la asignatura de Aplicaciones Web por Andrés Herranz, Axel Junestrand, Borja Lorenzo y Guillermo Rius. Consiste en una red social para facilitar la conexión entre personas con la finalidad de conocer gente nueva, pasarlo bien y mantenerse en forma.

La idea principal es crear una gran comunidad de personas que conecten de manera sencilla a través de nuestra página. Para ello, Pachanga se centra, en gran medida, en la creación de partidos de fútbol dónde las personas podrán jugar unas contra otras en los diferentes polideportivos de Madrid.

Además, para incentivar a nuestros usuarios, hemos creado un sistema de puntuación en función de tus victorias y derrotas que te permite luchar por los primeros puestos en el ranking de tu zona.

Nuestra aplicación web esta dirigida a todo tipo de personas de cualquier edad o nivel. Sólo tienen que tener una cosa en común: Actitud, ganas de aprender, ganas de participar y que les apasione el deporte y conocer a gente.

En Pachanga podrás registrarte de forma sencilla. Los usuarios registrados pueden ver los partidos que aún no se han disputado en su página principal y se pueden unir a ellos de manera sencilla. Además, tendrán la posibilidad de crear y gestionar nuevos partidos personalizados.

Actualmente, la mayor parte del consumo web se hace a través de dispositivos móviles como tablets o smartphones, y por eso Pachanga está desarrollada para poder ser utilizada con todo tipo de formato de pantalla gracias al diseño responsive.



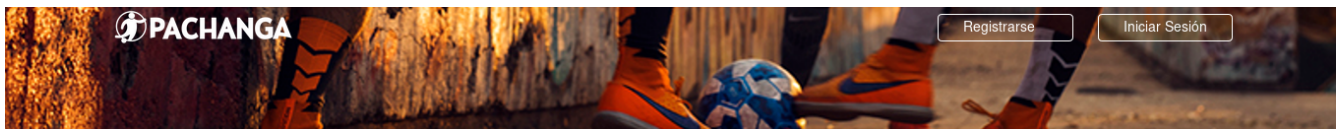
2 Pachanga

2.1 Página principal

Nuestro sitio web dispone de una primera página informativa en la que un usuario que aún no ha iniciado sesión podrá registrarse, loguearse y leer toda la información necesaria para conocer en qué consiste nuestra aplicación web y que funcionalidades tiene.

Para ello, la página principal se divide en tres secciones:

- Un header estático que le acompaña durante toda la navegación y le permite registrarse y loguearse.



- Una sección informativa de proyectos futuros a través de un slider dinámico, una breve introducción a lo que es Pachanga, las ideas principales de la web y un apartado para banners.



RegistrarseIniciar Sesión





Es una manera divertida de organizar pachangas de fútbol y conocer gente nueva.



Te mantiene en forma practicando tu deporte favorito.



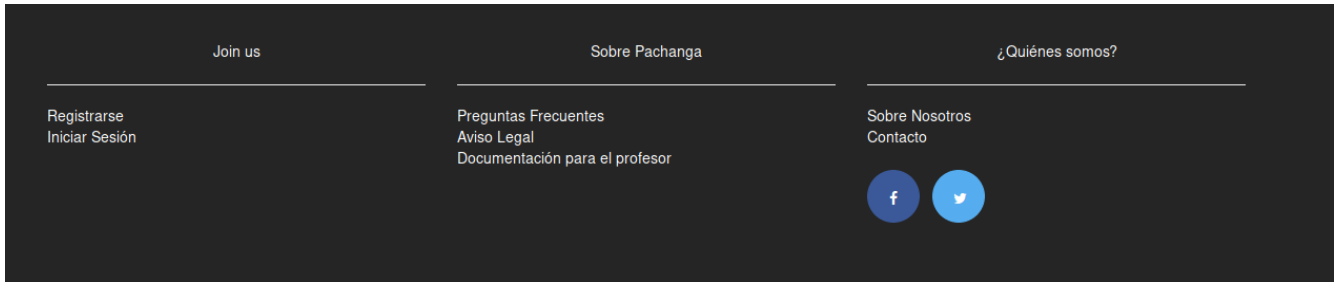
Es una buenísima forma de pasártelo bien con tus amigos.



Te permite mejorar como jugador y como persona, con un sistema de rating de usuarios.

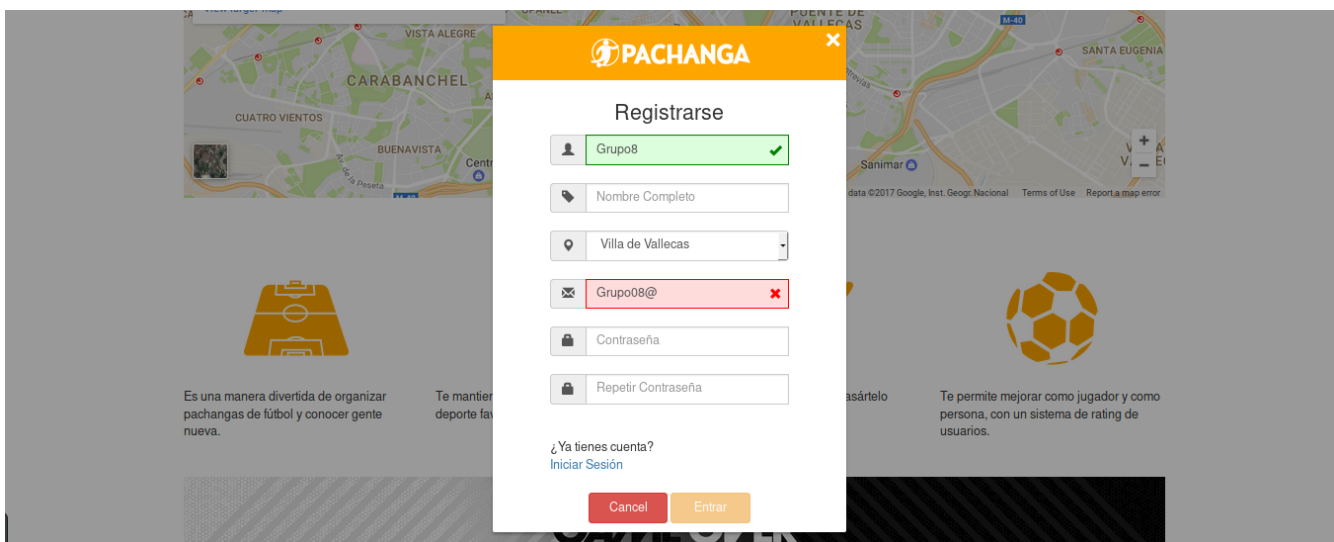


- Un pie de página o footer común a todas las páginas. En él dispondremos de acceso directo a otras secciones importantes de la web. Este apartado aparecerá cuando el usuario se registra pero se ocultarán los accesos al registro y al inicio de sesión.



2.1.1 Registrarse

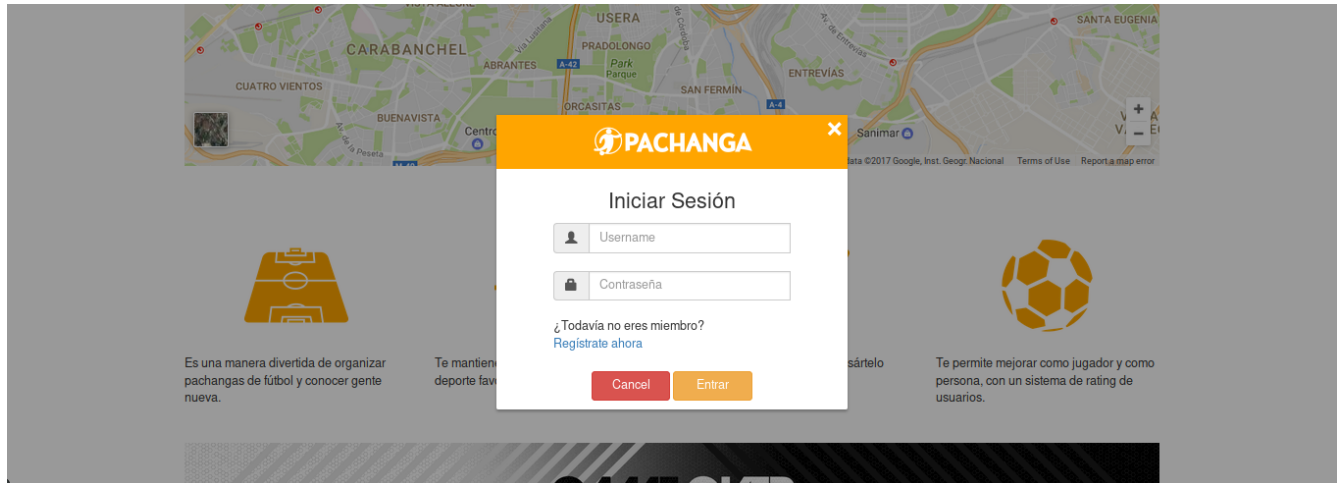
En cuanto al registro, será necesario definir nombre, nombre completo, distrito, email y contraseña. Disponemos de un validador (de lado del cliente) que comprueba que el nombre, el nombre completo y la contraseña tengan más de 5 caracteres, el email tenga la forma correcta y que las contraseñas sean iguales. Una vez cumplidos todos estos requisitos el botón de entrar se habilitará. En caso de error se redireccionará a otra página.





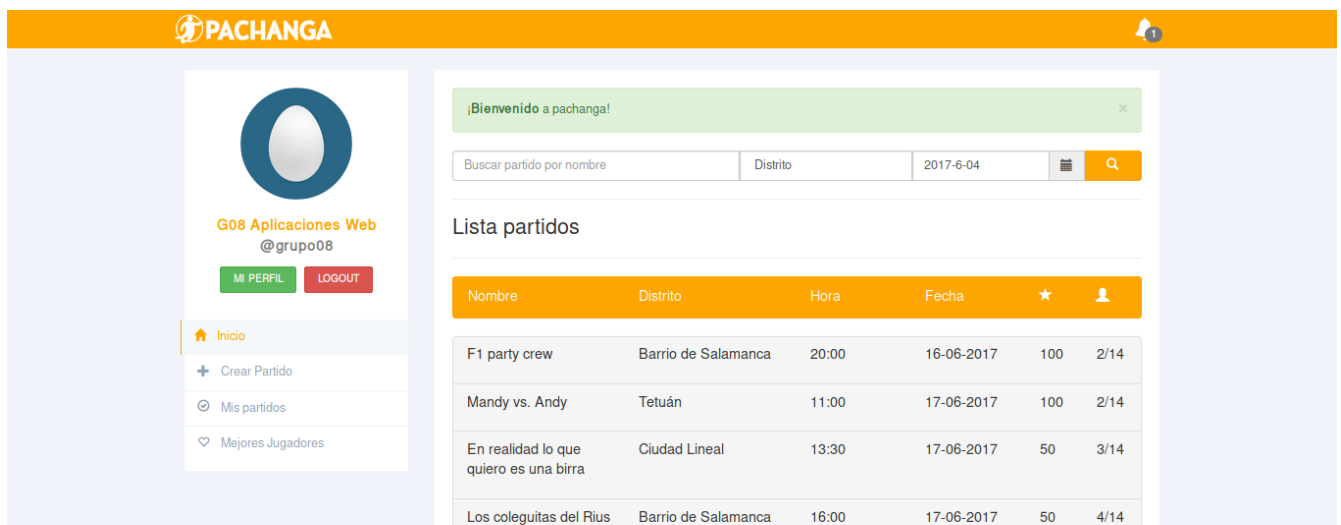
2.1.2 Iniciar Sesión

Si el usuario ya dispone de una cuenta, podrá acceder a través de su nombre de usuario y contraseña.



2.2 Página de inicio

Una vez iniciado sesión con éxito, seremos redireccionados a la siguiente página:





2.2.1 Diseño común

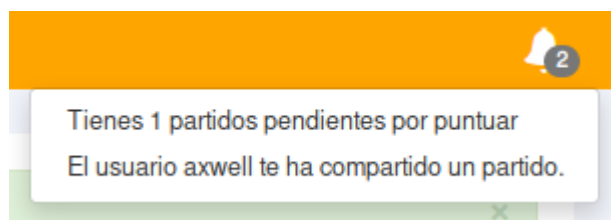
Para garantizar una buena navegación hemos seguido un diseño común a todas las páginas.

- En primer lugar, el header que permite en todo momento redireccionar a la página de inicio y muestra el panel de notificaciones.
- Por otro lado, el panel de usuario que muestra la información del usuario logueado, permite acceder a “mi perfil” y cerrar sesión. Además, permite navegar por las diferentes páginas de Pachanga.



2.2.2 Notificaciones

Utilizamos un sistema de notificaciones que nos alertará cuándo haya partidos creados por nosotros que se han pasado de fecha y que aún no hemos puntuado. Esta notificación nos redireccionará a la página de mi perfil para poder puntuarlos. Además, en la web hay habilitados botones que nos permiten compartir los partidos con nuestros amigos. Cuando alguien nos comparte un partido aparece una notificación que nos redirecciona a los datos del mismo.





2.2.3 Barra de búsqueda

Hemos implementado una barra de búsqueda que nos permite filtrar la lista de partidos según nuestros intereses. Nos posibilita buscar partidos por nombre, distrito y fecha.

Buscar partido por nombre	Distrito	01-06-2017		
---------------------------	----------	------------	--	--

2.2.4 Lista de partidos

A continuación, en la página de inicio podemos ver la lista de partidos que aún no se han jugado, ordenados por fecha. Esta lista muestra un resumen de los detalles de un partido. Cuando se selecciona uno se despliegan dos botones: Uno para poder ver los detalles del partido y otro que te permite compartir el partido.

Lista partidos

Nombre	Distrito	Hora	Fecha	★	👤
Informáticos vs Médicos	Barrio de Salamanca	13:11	04-06-2017	1333	1/14
<div>VerCompartir</div>					
F1 party crew	Barrio de Salamanca	20:00	16-06-2017	100	2/14
Mandy vs. Andy	Tetuán	11:00	17-06-2017	100	2/14
En realidad lo que quiero es una birra	Ciudad Lineal	13:30	17-06-2017	50	3/14
Los coleguitas del Rius	Barrio de Salamanca	16:00	17-06-2017	50	4/14



2.3 Mi perfil

En esta ventana podemos ver toda la información del usuario, además de todos los partidos creados por él. Si el partido se ha pasado de fecha podremos introducir el resultado del partido.



G08 Aplicaciones Web

-  **Username** grupo08
-  **E-Mail** grupo08@pachanga.com
-  **Distrito** Barrio de Salamanca
-  **Skill** 0
-  **Valoración** 0

Mis partidos

Informáticos vs Médicos	Barrio de Salamanca	13:11	03-06-2017	1333	1/14
		Ver	Puntuar		



2.3.1 Puntuar un partido

Una vez pulsado el botón de puntuar, se introducirá el resultado del partido y si nos ha gustado, y de esta forma se dará por concluido el encuentro. Automáticamente sumará 10 puntos a todos los integrantes del equipo ganador y restará 10 puntos a los integrantes del equipo perdedor.



Valorar: Informáticos vs Médicos

¿Cuál ha sido el resultado del partido?

¿Te ha gustado el partido?

☐ ☐

Enviar

2.4 Crear un partido

Los usuarios logueados tienen la posibilidad de crear un partido. Para ello elegirán nombre, fecha, hora, skill recomendado y distrito. Al escoger el distrito, se le habilitará la opción de elegir polideportivo para que pueda seleccionar uno dentro de los disponibles, así como un mapa con la localización del mismo.



Crear Partido

	Nombre del partido	
	04-06-2017	
	Hora	▲▼
	Minutos	▲▼
	Distrito	
	Skill recomendado	▲▼

Borrar

Enviar

Una vez rellenos los campos, si el nombre ya existe en la base de datos o la fecha es una fecha anterior a la actual, se mostrará un error al crear partido. Si todo ha salido bien, nos llevará a la página de detalles de un partido.

2.4.1 Detalles de un partido

En esta pestaña se mostrarán todos los datos de un partido, un mapa de la ubicación del polideportivo y nos permitirá unirnos a un partido siempre y cuando no nos hayamos unido ya y haya hueco en el partido. También podemos salirnos de un equipo. Cuando creamos un partido, automáticamente pasamos a formar parte del equipo 1.



Cuando un partido está finalizado, se mostrará la información correspondiente a los goles y no dejará que nos unamos.

Detalles del partido: Bootstrap lovers social match

Creador: grupo08
Distrito: San Blas
Polideportivo: Centro Deportivo Municipal San Blas
Skill: 456
Fecha: 04-06-2017
Hora: 15:00
Participantes:

Equipo 1	Equipo 2
ancelotti	andy
grupo08	axwell
maria	cr7
messi10	lorena
riusMiningMaster	mcardenas
superguidetti	-
-	-

[Jugar equipo 1](#) [Jugar equipo 2](#)



Detalles del partido: Bootstrap lovers social match

2 - 1

Partido Finalizado

Creador: grupo08
Distrito: San Blas
Polideportivo: Centro Deportivo Municipal San Blas
Skill: 456
Fecha: 04-06-2017
Hora: 13:00
Participantes:

Equipo 1	Equipo 2
ancelotti	andy
grupo08	axwell
maria	cr7
messi10	lorena
riusMiningMaster	mcardenas
superguidetti	-
-	-

[Jugar equipo 1](#) [Jugar equipo 2](#)





2.5 Mis partidos

Esta vista muestra todos los partidos en los que estamos apuntados, divididos en partidos jugados y partidos no jugados. Pulsando en la información podremos ir a los detalles del partido mencionados anteriormente.

Partidos no jugados

Nombre	Distrito	Hora	Fecha	★	👤
Informáticos vs Médicos	Barrio de Salamanca	13:11	03-06-2017	1333	1/14
Prueba horario	Ciudad Lineal	11:59	04-06-2017	200	1/14

Partidos jugados

Nombre	Distrito	Hora	Fecha	★	👤
Bootstrap lovers social match	San Blas	13:00	04-06-2017	456	11/14
Prueba horario 2	Moncloa-Aravaca	14:10	04-06-2017	300	1/14



2.6 Mejores jugadores

En la vista de mejores jugadores, el usuario podrá ver el ranking con los mejores jugadores de su zona. La lista estará ordenada por puntos de manera descendente.

Se mostrará también un mapa del distrito. Si el usuario logueado pulsa en el botón para ver el perfil de alguno de los usuarios del ranking, este podrá ver la información detallada de ese usuario. También podrá ver los partidos en los que esté registrado este usuario, obteniendo así un apartado social en la aplicación.

Mejores Jugadores de: Barrio de Salamanca

Jugador	Skill	
messi10	1010 ★★★	Ver Perfil
thespecialone	780 ★★★	Ver Perfil
superguidetti	610 ★★★	Ver Perfil
axwell	480 ★	Ver Perfil
grupo08	20 ★	Ver Perfil

Salamanca
[View larger map](#)

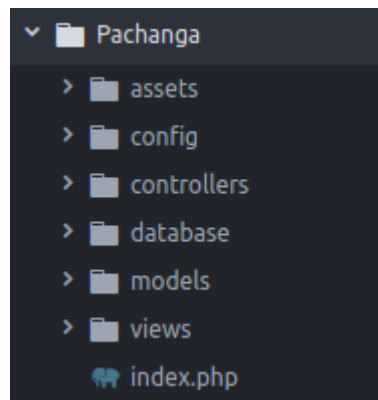


3 Arquitectura de la aplicación

3.1 Diseño

Para el desarrollo de nuestra aplicación hemos considerado oportuno utilizar como patrón principal de diseño el Modelo-Vista-Controlador (MVC) programado con orientación a objetos, puesto que creemos que es el que mejor representa la interacción cliente-servidor y permite una gran escalabilidad a nuestra página además de ser el más utilizado actualmente.

Hemos dividido nuestro proyecto en diferentes carpetas dónde cada una recoge las distintas funcionalidades del mismo:



- Assets: Esta carpeta engloba toda la parte de diseño (CSS) y Javascript de la que hablaremos más adelante, así como las imágenes y fuentes que se utilizarán en la web.
- Config: Aquí definimos las variables globales y las credenciales de la base de datos. Además, contiene la clase de conexión a la base de datos. Usa un patrón de diseño singleton ya que sólo conecta una vez a la base de datos: si esta conexión ya existe la devuelve y si no la crea.
- Database: Esta carpeta engloba lo relacionado con la base de datos y contiene un script que permite iniciar la base de datos e insertar algunos valores de prueba.



- **Modelo-Vista-Controlador:** Todos los enlaces de la web son redireccionados al index dónde parseará a qué tipo de controlador tiene que llamar y la acción que va a desempeñar. Si el controlador o la vista no existe, ejecutará la acción por defecto definida en la carpeta config.

- Models: En esta carpeta encontramos un modelo “padre” que, además de definir la conexión a la base de datos, define algunas funciones comunes a todos los modelos, como puede ser la búsqueda o el borrado.

Los otros modelos heredan de esta clase. Hemos definido un modelo por cada tabla de la base de datos.

Usamos el patrón de diseño DAO y es en estas clases donde se realizan las consultas a la base de datos (los insert, los delete o los update), para poder desvincular completamente el acceso a la base de datos con las vistas.

- Views: En este apartado es donde vamos a englobar todo lo relacionado con la impresión por pantalla. A partir de unos datos proporcionados por el controlador, los diferentes HTML se encargan de dar el contenido visual a la página.

Además, hemos definido (en la carpeta layouts) una serie de archivos PHP para recoger partes comunes de las diferentes vistas.

- Controllers: Hemos creado diferentes controladores para los principales módulos de la página que recogen funcionalidades similares. A través de la url y del index se creará un objeto controlador y desarrollará la acción que le sea enviada.

En estas clases se hace el control de usuarios, se conecta con el modelo para obtener información según la necesitemos y se llama a la vista correspondiente.



3.1.1 Ejemplo de uso (MVC)

Cuando queremos mostrar los mejores jugadores de un distrito, en primer lugar accedemos a esta url donde definimos el controlador y el tipo de acción que va a desarrollar:

<http://pachangafutbol.esy.es/index.php?controller=usuarios&action=mejores>

En el index.php se parseará y se ejecutará la acción, llamando al controlador “usuarios” y diciendo que queremos que ejecute el método “mejores”. Se creará una instancia del controlador definiendo una serie de atributos privados.

```
public function mejores(){

    $data = $this->usuario->getBy("id", $_SESSION["username"]);
    $mejores = $this->usuario->mejores($data[0]->getDistrito());
    $miDistrito = $this->distrito->getBy("id", $data[0]->getDistrito());
    $this->view("mejores", $this->entity, array(
        "data" => $data,
        "miDistrito" => $miDistrito,
        "mejores" => $mejores
    ));
}
```

Esta función recoge los datos del modelo a través de los métodos “getBy” y “mejores” que guarda en variables.

En el modelo hará la conexión a la base de datos y devolverá los datos en un array.

```
public function getBy($column, $value){
    $req = $this->db()->prepare("SELECT * FROM $this->table WHERE $column = :value");
    $req->execute(array('value' => $value));
    $result = $req->fetchAll(PDO::FETCH_CLASS, $this->class);
    return $result;
}
```



Una vez recogidos todos los datos, pasa a mostrarlos. Para ello se conecta con la vista a través del método “view” definido en el controlador “padre”, pasándole las variables necesarias.

Todos los datos que le llegan los convierte en variables, comprueba si el usuario puede acceder a la página correspondiente (si está logueado), actualiza las notificaciones y ejecuta la vista correspondiente.

```
public function view($vista, $entity, $datos){

    require_once('models/notificaciones.php');
    $this->notificacion = new Notificaciones();

    foreach ($datos as $id_assoc => $valor) {
        ${$id_assoc} = $valor;
    }

    if(isset($data)){
        $terminados = $this->notificacion->misFinalizados($data[0]->getId());
        $compartidos = $this->notificacion->misCompartidos($data[0]->getId());
    }

    require_once 'models/helperView.php';

    $helper = new HelperView();

    require_once 'views/'.$vista.$entity.'.php';
}

public function redirect($controlador=CONTROLADOR_DEFECTO,$accion=ACCION_DEFECTO){
    header("Location:index.php?controller=".$controlador."&action=".$accion);
}
```

La vista se encarga de pintar todos los datos que se le han pasado a través de HTML.

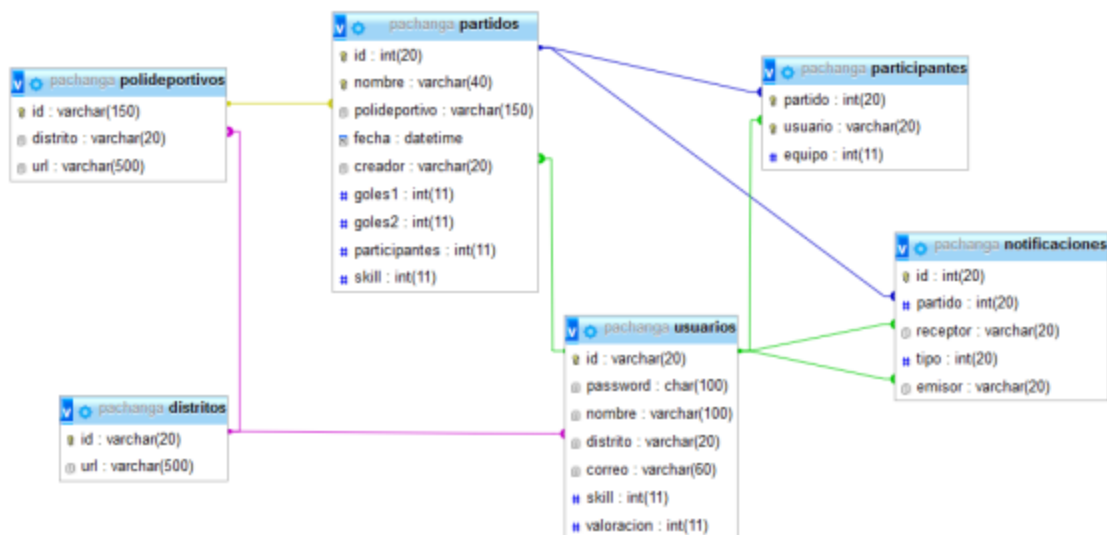


3.2 Base de datos

Para el diseño de la base de datos hemos utilizado la herramienta de phpMyAdmin que viene incluida en el paquete LAMPP, tal y como se vio en clase.

Hemos dividido la base de datos (“pachanga”) en seis tablas: “polideportivos”, “partidos”, “distritos”, “usuarios”, “notificaciones” y “participantes”, siendo esta última necesaria para representar la relación de muchos a muchos (N-N) que hay entre los usuarios y los partidos.

El diseño de la base de datos es el siguiente:



El acceso a la base de datos desde la aplicación se hace con la clase PDO (PHP Data Object), que es una interfaz ligera para poder acceder a bases de datos en PHP. Hemos decidido utilizar PDO porque permite la utilización de sentencias preparadas (PDOStatement), utilizadas para ejecutar la misma sentencia repetidamente con gran eficiencia y permite seguir con la arquitectura de orientación a objetos.



3.3 Decisiones de diseño

Como hemos comentado anteriormente, creemos que el diseño “mobile first” es un apartado muy importante en una aplicación como Pachanga. Para ello, hemos decidido utilizar Bootstrap como framework de diseño principal, ya que permite realizar webs responsivas.

Aunque hemos incluido la librería de Bootstrap, hemos modificado ciertas clases de esta desde otros archivos CSS sin modificar el archivo de Bootstrap. Como por ejemplo:

```
.glyphicon-ok{  
  display: none;  
  color: green;  
  margin-left: -22px;  
  margin-top: 10px;  
  /*margin-top: 3px;  
  margin-left: 235px;*/  
  position: absolute;  
  z-index: 32;  
}
```

```
.navbar-toggle {  
  height: 42.24px;  
  padding: 3px;  
  margin: 5px;  
  padding-left: 6px;  
  padding-right: 6px;  
}
```

Por otro lado, hemos elegido el naranja como color principal, ya que nos ha parecido llamativo y enérgico teniendo mucho que ver con Pachanga. También hemos intentado utilizar no más de 3 colores secundarios, siguiendo el patrón de diseño “Few hues many values”.

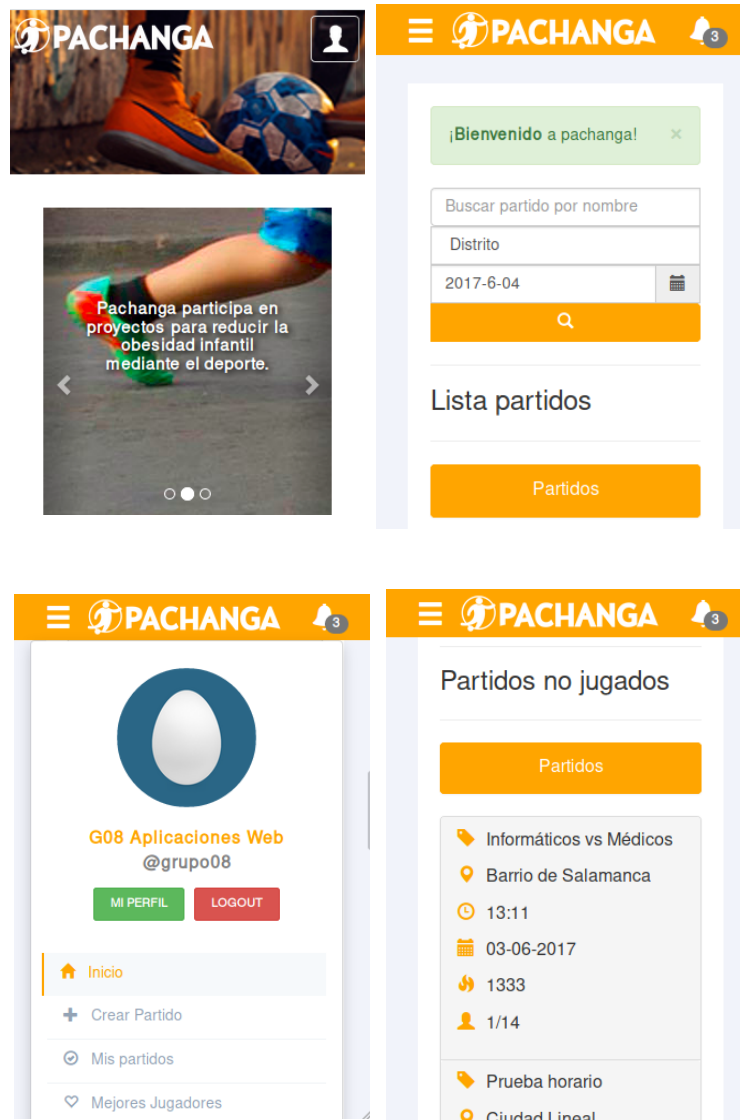
Para introducir animaciones a la página y evitar que fuera tan estática, hemos usado JQuery. Por ejemplo, en la página principal cuando haces scroll, la barra se convierte en naranja.

Nuestra idea principal ha sido que la navegación por nuestra web fuera sencilla, por lo que nos hemos decantado por un diseño minimalista para no recargar al usuario.



3.3.1 Diseño responsive

A continuación, vemos una serie de ejemplos de la web de Pachanga en su versión móvil, por ejemplo el botón de iniciar sesión, el panel de usuario desplegable, etc.





3.4 Otros apartados

3.4.1 Seguridad

- SQL Injection: Para proteger la página frente a ataques de SQL Injection, hemos saneado todos los parámetros proporcionados por el usuario antes de realizar las queries.

A parte del saneado, hemos utilizado las sentencias preparadas de PDO mencionadas en el apartado anterior. La característica principal de las sentencias preparadas es que la consulta y los datos se envían al servidor SQL de manera separada. Esto es importante, porque la base del ataque SQL Injection es aprovechar la mezcla de código y datos en las consultas SQL.

```
$hora = filter_var($_POST['hora'], FILTER_SANITIZE_NUMBER_INT);  
$minutos = filter_var($_POST['minutos'], FILTER_SANITIZE_NUMBER_INT);  
$skill = filter_var($_POST['skill'], FILTER_SANITIZE_NUMBER_INT);  
$polideportivo = filter_var($_POST['polideportivo'], FILTER_SANITIZE_STRING);  
$creador = filter_var($_SESSION['username'], FILTER_SANITIZE_STRING);
```

- Directory listing: Hemos impedido el listado de directorios mediante un fichero de configuración de apache (.htaccess).

```
Options -Indexes
```

- XSS (Cross Site Scripting): Para proteger a los usuarios de Pachanga frente a este tipo de ataques, cuando el usuario intenta acceder a alguna página mediante URL sin haber iniciado sesión, el sistema redirige a la página de home. Esto lo hemos hecho mediante una función de filtrado de la URL que redirige según convenga. Además, esto nos ha permitido ahorrar bastante código, ya que no hemos tenido que hacer las comprobaciones en todos los archivos, sino sólo en uno (“functions.php”).



```
function urlSinSesion(){
    session_start();
    if(isset($_SESSION['username'])){
        return false;
    }
    if(isset($_GET['controller'])){
        $contr = ucfirst($_GET['controller']);
    }else {
        return true;
    }
    if(isset($_GET['action'])){
        $act = ucfirst($_GET['action']);
    }else {
        return true;
    }
    if($_GET['action'] == 'login' || $_GET['action'] == 'register'){
        return false;
    }
    if($contr == 'View'){
        return false;
    }
    return true;
}
?>
```

- Encriptado de contraseñas: Todas las contraseñas son encriptadas antes de ser incluidas en la base de datos. Cuando el usuario inicia sesión, se hashea la contraseña proporcionada por este y se compara con la que hay en la base de datos. Nunca se descifran las contraseñas que hay en la base de datos.

El algoritmo utilizado para encriptar es SHA-512. La encriptación, paso por paso, es la siguiente:

- Se genera un string (\$salt) de 16 caracteres con md5 en base 64.
- Este string se hashea 10.000 veces (\$rounds=10000).
- Se encripta la contraseña con la función “crypt()”, que utiliza el algoritmo de encriptación SHA-512.

```
crypt(filter_var($_POST['password'], FILTER_SANITIZE_STRING), sprintf('$%$rounds-%d$%$', $rounds, $salt));
```



- **Fuerza Bruta:** La protección frente ataques de fuerza bruta se ha hecho incluyendo el RE-CAPTCHA de Google, para requerir confirmación si se proporciona erróneamente una contraseña, evitando así cualquier ataque de fuerza bruta a los formularios de inicio de sesión y registro.

3.4.2 AJAX y JSON

A la hora de crear un partido, en vez de elegir un polideportivo de una lista muy grande, hemos hecho que se cree dinámicamente el campo polideportivo según el distrito que se haya seleccionado anteriormente. Además, una vez elegido el polideportivo, cargamos la ubicación de dicho polideportivo a través de Google Maps. Para que no haya que recargar la página cada vez, hemos implementado estas opciones con AJAX que nos permite hacerlo sin recargar la página.

```
function mostrarMapa(polideportivo) {  
    if(polideportivo == "") {  
        document.getElementById("mapa").innerHTML = "";  
    } else {  
        if (window.XMLHttpRequest){  
            xmlhttp = new XMLHttpRequest();  
        } else {  
            xmlhttp = new ActiveXObject(Microsoft.XMLHTTP);  
        }  
  
        xmlhttp.onreadystatechange = function() {  
            if (this.readyState == 4 && this.status == 200) {  
                document.getElementById("mapa").innerHTML = this.responseText;  
            }  
        }  
  
        xmlhttp.open("GET","index.php?controller=partidos&action=mapa&polideportivo="+polideportivo, true);  
        xmlhttp.send();  
    }  
}
```




Por otro lado utilizamos el intercambio de ficheros JSON para poder validar el ReCaptcha implementado en la web. Los datos que te devuelve Google, una vez validado el ReCaptcha, vienen en formato JSON. Es necesario decodificar estos datos para ver si se ha validado correctamente.

3.4.3 Validadores y accesibilidad

Para evaluar la accesibilidad de Pachanga hemos sometido la web a diferentes validadores:

- TAW (www.tawdis.net) es una herramienta para el análisis de la accesibilidad de sitios web. Hemos elegido esta herramienta porque alcanza de una forma integral y global a todos los elementos y páginas que componen la web.

El análisis nos ha permitido comprobar que el nivel de accesibilidad (establecido por la WCAG) de nuestra página es AA.

Normativa	Resultado	Precedencia	Módulo de control
1.1.1 - Contraste de colores	✗	1	Contraste de colores
1.1.2 - Contraste de texturas	?	1	Contraste de texturas
1.2.1 - Línea del color	?	1	Línea del color
1.2.2 - Contraste de texturas	?	1	Contraste de texturas
1.3.1 - Línea del color	?	1	Línea del color
1.3.2 - Contraste de texturas	?	1	Contraste de texturas
1.4.1 - Línea del color	?	1	Línea del color
1.4.2 - Contraste de texturas	?	1	Contraste de texturas
1.4.3 - Línea del color	?	1	Línea del color
1.4.4 - Contraste de texturas	?	1	Contraste de texturas



- PowerMapper (www.powermapper.com) es otra herramienta para comprobar la accesibilidad de la página.

The screenshot shows the PowerMapper interface with a green checkmark logo and the text "one click website tools". The navigation bar includes links for Home, Products, Buy, Support, and About Us. Below the navigation bar, there are tabs for Summary, Issues, and Pages. The Summary tab is selected, showing a dashboard with various categories and their corresponding issues. The categories include Overall Quality, Errors, Accessibility, Compatibility, Search, Standards, Usability, and Totals. Each category has a bar chart indicating the number of issues and a benchmark comparison. The benchmark comparison shows that 11% of sites have issues, worse than average, while 0% have issues, better than average. The overall quality is 1 page with quality issues, 0 pages with broken links or other errors, 1 page with accessibility problems, 1 page with browser specific issues, 0 pages with search engine issues, 1 page with W3C standards issues, and 1 page with usability issues. The total number of pages and files checked is 10. A note at the bottom states: "The trial version is limited to checking 10 pages and images."

Category	Issues	Pages	Benchmark
Overall Quality	1	1 pages with quality issues	11% have issues, worse than average
Errors	0	0 pages with broken links or other errors	0% have issues, better than average
Accessibility	1	1 pages with accessibility problems	11% have issues, better than average
Compatibility	1	1 pages with browser specific issues	11% have issues, worse than average
Search	0	0 pages with search engine issues	0% have issues, better than average
Standards	1	1 pages have W3C standards issues	11% have issues, better than average
Usability	1	1 pages with usability issues	11% have issues, better than average
Totals		10 pages and files checked	

The trial version is limited to checking 10 pages and images.

Esta página mostraba varios errores:

The screenshot shows a red error icon and the text "Duplicate attribute 'class'". The error is located in the HTML5 file at line 161. The URL is <http://pachangafutbol.esy.es/index.php?controller=View&action=home>. The error affects 1 page.

Este error lo hemos tenido que dejar por un tema de estética en el login, ya que los inputs quedaban descentrados si se ponían en el mismo "".

The screenshot shows a red error icon and the text "All ONCLICK handlers should have an equivalent ONKEYPRESS handler." The error is located in the HTML5 file at line 185. The URL is <http://pachangafutbol.esy.es/index.php?controller=View&action=home>. The error affects 1 page. The error message also includes a reference to Section 508 (2000) 1194.22 (i), WCAG 2.0 A FS4, and Section 508 (2017) A FS4.



Este error bajaba el nivel de accesibilidad de la página ya que, como bien explica, algunos usuarios no pueden utilizar un ratón. Lo hemos resuelto añadiendo, además de “onclick()”, “onkeypress()”, para que la ventana se pueda abrir o cerrar pulsando una tecla.

También hemos pasado los validadores de código:

- Para validar el código de nuestra aplicación, hemos hecho uso del un HTML Checker (<https://validator.w3.org/nu>), pegando los links a las distintas vistas de la página hasta que desaparecieran todos los errores.

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for <http://pachangafutbol.esy.es/>

Checker Input

Show ☐ source ☐ outline ☐ image report

Check by

Document checking completed. No errors or warnings to show.

Used the HTML parser. Externally specified character encoding was UTF-8.
Total execution time 514 milliseconds.

- Del mismo modo, W3 ofrece un validador de CSS3. En este caso, aparecen una serie de errores que no podemos corregir, ya que las propiedades permiten animaciones y cambios de estilo.

Disculpas! Hemos encontrado las siguientes errores (1)	
URI : <i>index.css</i>	
191	Lo lamentamos, la regla-arroba @-webkit-keyframes no está implementada.



3.4.4 Search Engine Optimization (SEO)

Para mejorar la indexación de la página en los distintos buscadores (Google, Bing, etc.), hemos generado un mapa del sitio (sitemap) con la herramienta de <https://xmlsitemapgenerator.org>, que crea un archivo xml con el mapa del sitio, sin añadir las páginas que requieren login.

La URL del sitemap es <http://pachangafutbol.esy.es/sitemap.xml>

4 Instrucciones de uso

Lo primero que debe hacer el usuario para utilizar Pachanga es registrarse en la aplicación, escogiendo el distrito al que pertenece, pudiendo elegir entre los 21 distritos de Madrid.

Una vez registrado, podrá ver una página de inicio con todos los partidos que hay registrados, y filtrar los mismos por nombre, distrito y fecha.

Tendrá, además, un menú en la parte lateral izquierda (o pulsando el menú Burger en la versión responsive), desde el que podrá crear un partido, ver sus partidos registrados, y ver los mejores jugadores de su distrito (siempre con la opción de volver a la página de inicio pulsando el logo de Pachanga o seleccionando “Inicio” en el menú).

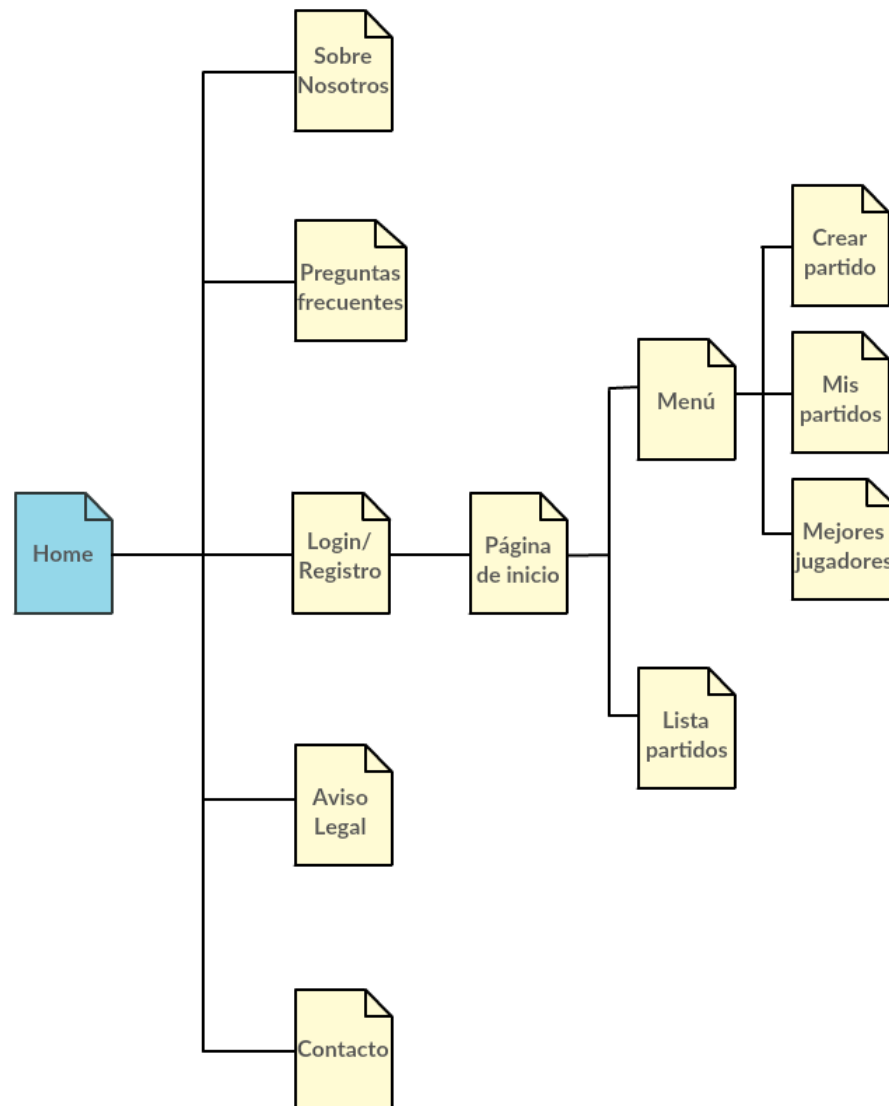
El usuario puede interactuar con otros compartiendo los partidos que desee con los demás usuarios de la aplicación.

Cuando los partidos registrados por el usuario hayan acabado (cuando se pase la fecha), pasarán a formar parte de los partidos jugados, y se enviará una notificación al usuario informando al mismo de que puede puntuarlo. Cuando lo puntúa, el resto de los usuarios que han jugado este partido podrán ver el resultado del mismo.



4.1 Site Map

La imagen a continuación representa la estructura de la página:





5 Conclusión

Hemos desarrollado la práctica utilizando el Modelo-Vista-Controlador, tal y como se hace en proyectos reales. Con ello, hemos podido ver las ventajas y desventajas que esto conlleva.

Una desventaja sería la dificultad del desarrollo y el entendimiento del código en su plenitud al inicio del proyecto. Pero esto queda compensado con ventajas tales como: la organización del código, el control de errores, la escalabilidad y la posibilidad de añadir nuevas funcionalidades a la aplicación.

Como conclusión, nos ha parecido muy interesante tomarnos esta práctica como si fuera un proyecto real y enfrentarlos a las diferentes dificultades que han ido surgiendo a lo largo del proceso.

Además, el desarrollo de esta aplicación nos ha permitido investigar y conocer en profundidad las distintas tecnologías aplicadas actualmente en el desarrollo web, como pueden ser HTML, CSS, JQuery, AJAX o Bootstrap.