



# PRESENTACIÓN TÉCNICA

Adrián Postigo

# Evolución de las funcionalidades durante los Sprints

- En un estado base solo se realizó la parte de front-end, es decir a nivel visual y su correspondiente redirección a la paginas pertinentes
- Una vez diseñada e implementada la base de datos se realiza una de las primeras funcionalidades, el login
- Implementado el login se procede a incorporar la creación de nuevos usuarios
- Una vez finalizados las funcionalidades anteriores se empieza con el diseño de los productos y el perfil del producto
- Finalmente la creación de listas y permitir añadir productos
- Como ampliación se ha diseñado la última funcionalidad que permite añadir amigos y ver su lista de deseos

## Problemas y soluciones

- El principal problema al que me he enfrentado fue en la gestión de errores ya que el navegador daba muy poca información en relación a este.
  - o Los métodos para solucionar los errores han sido emplear try catch para conexión a la BD, la función "var\_dump()" que me permitía saber que había dentro de la variable sobre todo para saber si los datos se estaban pasando, finalmente comentando líneas de código para saber dónde fallaba todo.
- El segundo problema remarcable fue en la fase de despliegamiento
  - o La solución a este problema fue darle permisos de escritura a la carpeta cache mediante el programa "Putty".
- Otro problema remarcable fue pasar información de la vista al controlador
  - o Se solucionó empleando :

```
- "http://$_SERVER[HTTP_HOST]$_SERVER[REQUEST_URI]"
```
- Por ultimo tuve problemas con la permanencia de la sesión:

- Se solucionó añadiendo al archivo index.php la función “sesión\_start”

## Aspectos técnicos

Framework seleccionado: Codeigniter 4.0

Codeigniter es un framework para aplicaciones web escrito en PHP y de código abierto bajo licencia Apache License. En la versión 3, la licencia será Open Software License (OSL). Este entorno de trabajo está especialmente escrito para crear sitios webs dinámicos de manera sencilla y rápida.

Lenguaje escogido: PHP

PHP es un lenguaje de programación de uso general que se adapta especialmente al desarrollo web.

## Explicación técnica de las funcionalidades

Login:

```
$username = $_POST['uname'];  
$pass = $_POST['psw'];  
$encrypted_pass = md5($pass);  
echo($encrypted_pass);
```

Recuperamos por método POST tanto el email como el password, encriptamos la contraseña y se compara con la contraseña encriptada en la BD

```
//validation password  
foreach ($query->getResult('array') as $row)  
{  
    if (strcmp($row['Password'], $encrypted_pass)==0){  
        $okpass =1;  
    }  
}
```

Validamos que la contraseña y el email sean correctos y se inicia la sesión. Se comprueba que el usuario no es el administrador

Al mismo tiempo obtenemos los datos del usuario.

Para obtener la lista de deseos, al tener como identificador único de la lista el propio email del usuario.

```
//Wishlist
$sql = "SELECT Items FROM wishlist WHERE UserMail = ?";
$query = $db->query($sql, $_SESSION['email']);

foreach ($query->getResult('array') as $row)
{
    $items = $row['Items'];
}

$array_items = str_split($items);

//HERE
foreach ($array_items as $char) {
    if($char != ','){
        $sql = "SELECT Name FROM product WHERE Img = ?";
        $query = $db->query($sql, $char);
        foreach ($query->getResult('array') as $row)
        {
            array_push($products_names, $row['Name']);
        }
    }
}
```

Se localiza la lista y se extraen los productos. En la lista los productos se guardan por el número de la imagen, se obtienen los identificadores y se busca el nombre del producto.

Todo esto se envía a la vista del perfil y se redirecciona hacia allí.

### Nuevo Usuario:

Se emplea el mismo criterio que en el login el cual es la funcionalidad antes mencionada.

```
//Registro
$query = $db->query("INSERT INTO user (Name, Surname, Age, Email, Password) VALUES ('$uname', '$sname', '$age', '$mail', '$encrypted_pass')");

//Create de wishlist
$query = $db->query("INSERT INTO wishlist (UserMail) VALUES ('$mail')");

//Create de friendlist
$query = $db->query("INSERT INTO friendlist (Owner) VALUES ('$mail')");

echo view('/inc/header');
echo view('home');
}else{
    echo view('/inc/header');
    echo view('signin');
}
```

Se insertan los datos en la tabla User y se aprovecha para crear la lista de deseos junto a la lista de amigos.

### Products:

```
$query = $db->query("SELECT Img FROM product");

foreach ($query->getResult('array') as $row)
{
    array_push($products, $row['Img']);
}

$data['products'] = $products;
```

Se guardan los números de los productos creados y se envían a la vista productos.

```
<div class="container col-sm-12">
    
    <button class="btn" role="link" onclick="window.location='product?arg=<?php echo $row?>'">Profile</
    button>
    <?php
    if (isset($_SESSION['name'])) {?>
        <button class="btn" role="link" onclick="window.location='wish?arg=<?php echo $row?>'">Wish</button>
    <?php }?>
</div>
<?php
}
?>
```

Utilizando PHP y html se hace un bucle que vaya printando los contenedores.

### Product:

```
$product_info = array();

$actual_link = "http://$_SERVER[HTTP_HOST]$_SERVER[REQUEST_URI]";

$img_id = substr($actual_link, -1);
```

Para poder enviar información al controlador se coge la URL y se guarda el último valor que es el número de la imagen y la misma vez lo que se usa como identificador único.

```
$sql = "SELECT * FROM product WHERE Img = ?";
$query = $db->query($sql, $img_id);

foreach ($query->getResult('array') as $row)
{
    array_push($product_info, $row['Name'] );
    array_push($product_info, $row['Price'] );
    array_push($product_info, $row['Description'] );
    array_push($product_info, $row['Link'] );
}
$img_url = "../public/Assets/img/".$img_id.".jpg";
array_push($product_info,$img_url );
$data['product_info'] = $product_info;

echo view('/inc/header');
echo view('product', $data);
```

Se obtiene toda la información del producto y se envía toda la información al perfil del producto.

### Perfil del Usuario:

Se emplea la misma estructura que en el perfil del producto pero esta vez se usa como dato el email de la sesión activa, se recoge toda la información del usuario y también de su lista de deseos y se envía a la vista.

En la vista se implementa el botón para ver todos tus amigos y de ahí poder acceder a sus datos.

## Panel de Administrador:

```
if (strcmp($_SESSION['email'], "admin@gmail.com") == 0){  
    $data['ok'] = "ok";  
    echo view('/inc/header');  
    echo view('admin');  
}else{  
    $data['ok'] = "ko";  
    echo view('/inc/header');  
    echo view('login',$data);  
}
```

Como medida de seguridad se implementa una condición que compruebe previamente si la sesión iniciada es la de administración, de no ser el caso no permite que se acceda a ella por la URL.

Se añaden tres funciones complementarias que permiten crear productos, borrarlos y borrar usuarios con el mismo procedimiento mencionado en funcionalidades anteriores.

## Función Añadir Deseo:

Se recupera de la URL el número del producto y se añade a la lista de la sesión activa.

```
foreach ($query->getResult('array') as $row)  
{  
    $items = $row['Items'];  
}  
  
$array_items = str_split($items);  
  
foreach ($array_items as $char) {  
    if ($char == $img_id){  
        $exists = 1;  
    }  
}
```

Se comprueba que el producto no exista en la lista del usuario y se concatena con una “,” para facilitar la búsqueda de los productos posteriormente.

### Función buscar amigos:

```
$sql = "SELECT Name FROM user WHERE Name LIKE ?";  
  
$query = $db->query($sql, $user_to_search."%");  
  
foreach ($query->getResult('array') as $row)  
{  
    array_push($users, $row['Name']);  
}
```

Implementando el like y el tanto por ciento se consigue obtener los usuarios con un nombre parecido al introducido, también se selecciona el email y se envía a la vista.

### Función añadir amigo:

```
$array_friends=array();  
$actual_link = "http://$_SERVER[HTTP_HOST]$_SERVER[REQUEST_URI]";  
$exists = 0;  
  
//$user_to_follow = substr($actual_link, -1);  
  
$pos = strpos($actual_link, "=");  
$len = strlen($actual_link);  
  
$pos++;  
$subr = $len - $pos;  
  
$user_to_follow = substr($actual_link, -$subr);
```

Se recupera la URL y se calcula la diferencia entre el final de la URL y el símbolo "=", así se obtiene el email.

Se comprueba que el amigo no exista previamente y el valor se guarda en una variable para avisar al usuario ya había sido añadido previamente.

Obtenido el dato se añade a la lista y se concatena con "," para diferenciarlos.

### Ver lista de Amigos:

Se recupera el email se la lista de amigos y con él se recuperan el nombre de este.



## Ver Perfil de Amigos:

Se recupera el identificador del amigo de la misma manera que en la función añadir amigo se selecciona toda la información y se envía a la vista.

## Estructura de la Base de Datos:

### Tabla Usuario:

+ Opciones		ID	Name	Surname	Age	Email	Password
<input type="checkbox"/>	Editar Copiar Borrar	3	admin	admin	0	admin@gmail.com	c93ccd78b2076528346216b3b2f701e6
<input type="checkbox"/>	Editar Copiar Borrar	4	papus	the great	27	papus@gmail.com	f39d9a53f98e528d573ab4388bafd3b9
<input type="checkbox"/>	Editar Copiar Borrar	6	pepe	seguí	22	pepe@gmail.com	5e783f68bbe280088d77a82cbd235a0d
<input type="checkbox"/>	Editar Copiar Borrar	7	pepe99	p	22	pepe99@gmail.com	7fbc653e835335c318a6eb574f58c163
<input type="checkbox"/>	Editar Copiar Borrar	8	sandro	uskus	22	sandro@gmail.com	7fbc653e835335c318a6eb574f58c163
<input type="checkbox"/>	Editar Copiar Borrar	9	usuario	user	30	usuario@gmail.com	781c5b5f7e460328d3871ddda9468fda














### Tabla Producto:

+ Opciones		ID	Name	Price	Description	Link	Img
<input type="checkbox"/>	Editar Copiar Borrar	1	Red Dead Redemption 2	65	Juego de acción del oeste	<a href="https://www.rockstargames.com/es/">https://www.rockstargames.com/es/</a>	1
<input type="checkbox"/>	Editar Copiar Borrar	2	Collar MKzenie	25	Collar de plata	<a href="https://www.tous.com/es-ca/">https://www.tous.com/es-ca/</a>	2
<input type="checkbox"/>	Editar Copiar Borrar	3	Xiaomi 3Q	200	Mobil de última generación	<a href="https://www.mi.com/es/">https://www.mi.com/es/</a>	3
<input type="checkbox"/>	Editar Copiar Borrar	4	Anillo Luxury	100	Anillo de plata y diamante	<a href="https://www.aristocrazy.com/es/joyeria/anillos/">https://www.aristocrazy.com/es/joyeria/anillos/</a>	4

### Tabla Lista de Deseos:

		ID	WishListName	UserMail	Items
<input type="checkbox"/>	Editar Copiar Borrar	2		admin@gmail.com	
<input type="checkbox"/>	Editar Copiar Borrar	3		papus@gmail.com	,1,2
<input type="checkbox"/>	Editar Copiar Borrar	5		pepe@gmail.com	,1
<input type="checkbox"/>	Editar Copiar Borrar	6		pepe99@gmail.com	,3
<input type="checkbox"/>	Editar Copiar Borrar	7		sandro@gmail.com	,1
<input type="checkbox"/>	Editar Copiar Borrar	8		usuario@gmail.com	

## Tabla Lista de Amigos:

				ID	Owner	FriendNames
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1	pepe@gmail.com	papus@gmail.com, pepe99@gmail.com, pepe99@gmail.com,
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	2	pepe99@gmail.com	papus@gmail.com, pepe@gmail.com, usuario@gmail.com,
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	3	sandro@gmail.com	pepe@gmail.com, pepe99@gmail.com, papus@gmail.com,
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	4	usuario@gmail.com	