

Proyecto:

# **Creación de Tienda Virtual**

# Índice

Introducción.....	2
Requisitos Mínimos.....	3
1. Página de Bienvenida.....	3
2. Formulario de Registro.....	4
3. Procesamiento del Formulario.....	5
4. Página de Inicio de Sesión.....	8
5. Estilo.....	11
6. Página de consultas.....	12
7. Añadir medidas de seguridad.....	13
8. Extra.....	14
8.1. Carrito.....	14
8.2. Aplicaciones de Ofimática Web.....	15
8.3. Información Adicional.....	15

# Introducción


El proyecto se trata de una tienda virtual de suplementos deportivos enfocado al uso de clientes, la tienda se llama MuscleBoost Palace y se compone de:

- Una página de bienvenida en la cual podemos ver los productos, con su nombre, foto y precio, una columna de categorías, la cual nos permite filtrar los productos por categorías, un botón que funciona como enlace a una página que realiza dos consultas a la base de datos, un botón que nos muestra un formulario para registrarnos, otro que nos permite iniciar sesión y finalmente un icono de carrito que nos lleva a nuestro carrito.
- En la página de la consulta podemos ver cuántos usuarios registrados hay en la base de datos y cuantas compras se han realizado. Además hay un enlace a un formulario realizado con Google Forms desde el cual el usuario podrá dar su opinión de la página
- En la página del carrito podemos añadir o quitar productos, seleccionar el producto y la cantidad y se nos mostrará el precio de cada artículo multiplicado por la cantidad al final de la línea, aparte también se muestra el precio total de la compra, sumando el precio total de todos los productos. Debajo del precio total tenemos un botón que nos permite realizar la compra y enviar los datos de esta a la tabla compras en nuestra base de datos.

# Requisitos Mínimos


## 1. Página de Bienvenida


**MBP**  
MuscleBoost Palace


 [Iniciar Sesión](#) [Registrarse](#)


**Categorías**  
[Mostrar todo](#)  
[Nutrición Deportiva](#)  
[Alimentación](#)  
[Accesorios](#)


¡Consulta los datos de nuestra pagina web!


  
**Proteína - 1Kg**  
59,99€

  
**Proteína - 500g**  
29,99€

  
**Shaker - 500ml**  
3,99€

  
**Shaker - 700ml**  
5,99€

  
**Barrita Energética 1**  
1,99€

  
**Barrita Energética 2**  
2,99€

Como se comentó en la introducción, esta es la página principal, como se puede ver tiene un header que incluye el logo de la tienda a la izquierda y los enlaces al carrito, al formulario de inicio de sesión y al de registro. Debajo del header encontramos el contenido de la tienda que se compone de una columna de categoría desde la cual podemos filtrar los artículos:

**Categorías**  
[Mostrar todo](#)  
[Nutrición Deportiva](#)  
[Alimentación](#)  
[Accesorios](#)

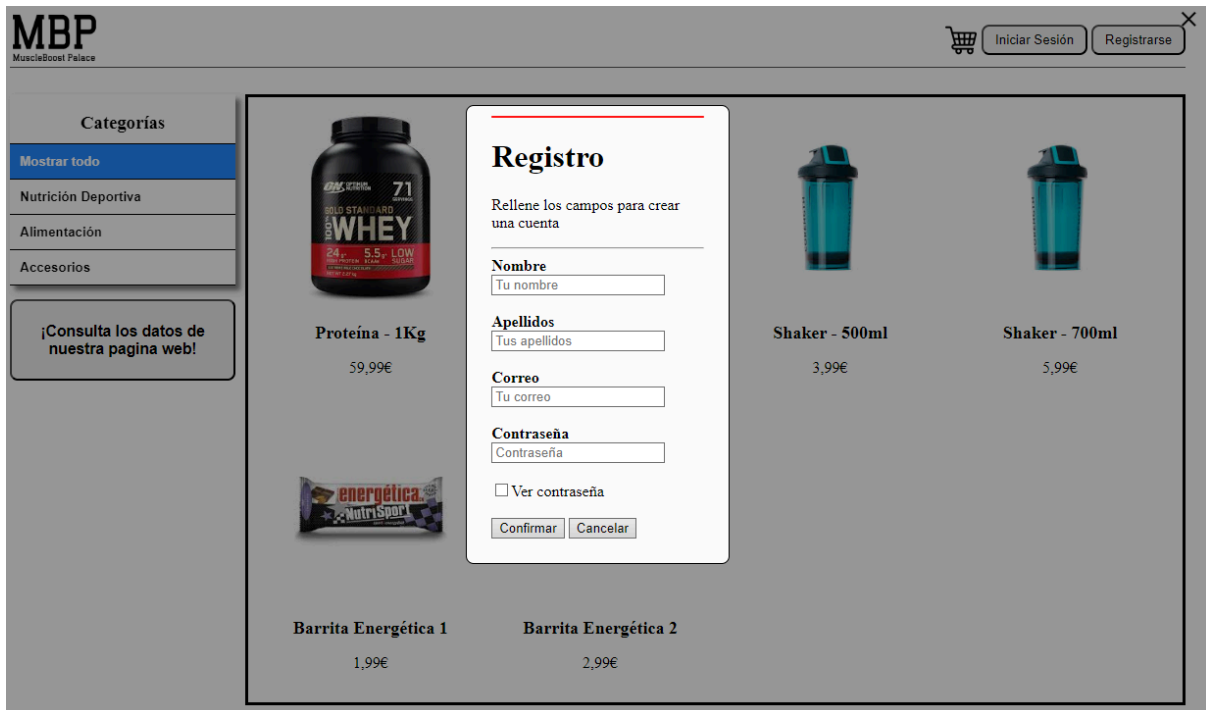
¡Consulta los datos de nuestra pagina web!

  
**Shaker - 500ml**  
3,99€

  
**Shaker - 700ml**  
5,99€

Debajo de esta columna tenemos un botón que nos lleva a la página de la consulta realizada a la base de datos, la cual se mostrará más adelante.

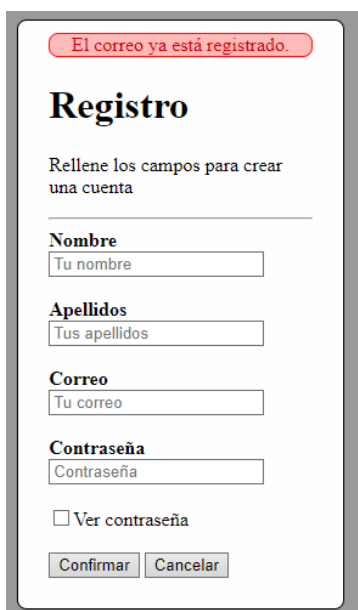
## 2. Formulario de Registro



The screenshot shows the MBP (MuscleBooel Palace) website with a registration form overlay. The website header includes the MBP logo, a shopping cart icon, and buttons for 'Iniciar Sesión' and 'Registrarse'. The left sidebar lists categories: 'Mostrar todo', 'Nutrición Deportiva', 'Alimentación', and 'Accesorios'. The main content area displays various products: 'Proteína - 1Kg' (59,99€), 'Shaker - 500ml' (3,99€), 'Shaker - 700ml' (5,99€), 'Barrita Energética 1' (1,99€), and 'Barrita Energética 2' (2,99€). The registration form is titled 'Registro' and prompts the user to 'Rellene los campos para crear una cuenta'. It includes input fields for 'Nombre', 'Apellidos', 'Correo', and 'Contraseña', along with a checkbox for 'Ver contraseña' and 'Confirmar'/'Cancelar' buttons.

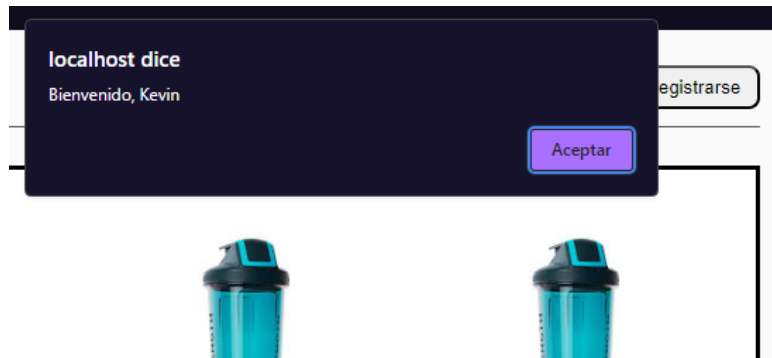
Este es el formulario de registro, como se puede apreciar se muestra sobre la página principal. Esto hace que para realizar su procesamiento haya que hacer algunos cambios en nuestro fichero que no haríamos en caso de tener el formulario en una página a parte y su procesamiento en otra. En este caso el código del formulario se encuentra en el mismo fichero que el código de la página principal, gracias al uso de modales podemos hacer que esté oculto y que solo se muestre cuando pulsemos el botón.

Cuando queramos registrar un usuario en la base de datos tan solo tenemos que rellenar los campos del formulario y pulsar confirmar. Si el correo ya se encuentra en la base de datos se nos mostrará el mensaje que aparece en la imagen.



This image is a close-up of the registration form, showing an error message at the top: 'El correo ya está registrado.' (The email is already registered). The form is titled 'Registro' and prompts the user to 'Rellene los campos para crear una cuenta'. It includes input fields for 'Nombre', 'Apellidos', 'Correo', and 'Contraseña', along with a checkbox for 'Ver contraseña' and 'Confirmar'/'Cancelar' buttons.

En cambio, si el correo no está registrado nos llevará a una página idéntica a la página de bienvenida pero con una url distinta, esto se hace así para que los datos del formulario no se queden guardados y nos muestre el error de que el usuario está ya registrado, ya que si registramos un usuario, lo añadimos a la base de datos, volvemos a la página principal y recargamos la página, este formulario se volvería a enviar causando que se muestre dicho error. Al registrar una cuenta correctamente se envían los datos a un script que procesa el contenido del formulario y añade la información del usuario a la tabla Usuarios en nuestra base de datos y muestra el siguiente mensaje:



En el formulario de registro podemos activar o desactivar la visibilidad de la contraseña:

<b>Contraseña</b> <input type="password" value="...."/>	<b>Contraseña</b> <input type="text" value="12345"/>
<input type="checkbox"/> Ver contraseña	<input checked="" type="checkbox"/> Ver contraseña

### 3. Procesamiento del Formulario

Para poder procesar el formulario e introducir los datos recibidos en una tabla, primero tenemos que crear una base de datos luego las tablas y por último introducir los datos del formulario. La solución que he optado por hacer es que cada vez que se acceda a la página principal se cree una conexión al servidor, en este caso phpMyAdmin, y cree la base de datos en caso de que no esté creada, cerramos esta conexión y abrimos otra seleccionando la base de datos, haciendo esto nos aseguramos que la base de datos esté creada cuando vayamos a crear las tablas, ya que en caso de que realizáramos la primera conexión incluyendo la base de datos, si ésta no estuviera creada, daría un error, así que mejor asegurarse. Una vez realizada la segunda conexión con la base de datos seleccionada creamos las tablas en caso de que no existan. En la siguiente imagen se muestra la segunda conexión y la creación de las tablas.

```
// Nueva conexion con la base de datos asegurada
$conn = mysqli_connect($servername, $username, $password, $dbname);

//Verificamos de nuevo la conexion
if (!$conn) {
    die("La conexión falló: " . mysqli_connect_error());
}

// Creacion tablas en la base de datos, primero la seleccionamos
mysqli_select_db($conn, "MuscleBoost");

// Creamos las tablas
$sqlCrearTablaUsuarios = "CREATE TABLE IF NOT EXISTS Usuarios (
    nombre varchar(15) not null,
    apellidos varchar(50) not null,
    correo varchar(100) primary key,
    contrasena varchar(255) not null
)";

if (mysqli_query($conn, $sqlCrearTablaUsuarios)) {
    echo "";
} else {
    echo "Error al crear la tabla: " . mysqli_error($conn);
}

$sqlCrearTablaCompras = "CREATE TABLE IF NOT EXISTS Compras (
    id_compra int(6) UNSIGNED AUTO_INCREMENT primary key,
    /*correo varchar(100) not null,*/
    producto varchar(50) not null,
    cantidad int not null,
    precioTot double not null
    /*constraint fk_compra_usuarios foreign key (correo) references Usuarios(correo)*/
)";

if (mysqli_query($conn, $sqlCrearTablaCompras)) {
    echo "";
} else {
    echo "Error al crear la tabla: " . mysqli_error($conn);
}
}
```

Como se puede apreciar en la imagen de arriba, en la tabla Compras hay dos líneas comentadas, las cuales, si estuvieran sin comentar, incluirían una columna de correo que haría referencia al correo del usuario y por tanto se podría relacionar la compra con el usuario que la realizó.

A continuación se mostrará el código que procesa la información del formulario de registro:

```
// if para que solo se ejecute al hacer submit
if ($_SERVER["REQUEST_METHOD"] == "POST" && isset($_POST['registro'])) {

    // Añadir Usuario mediante registro comprobando si ya está en la tabla o no
    $nombre = $_POST['nombre'];
    $ape = $_POST['ape'];
    $correo = $_POST['correo'];
    $contra = password_hash($_POST['contra'], PASSWORD_DEFAULT);

    $comprobarExiste = "SELECT correo FROM Usuarios WHERE correo = '$correo'";
    $resultado = mysqli_query($conn, $comprobarExiste);

    //Si el resultado es mayor que 0, está en la tabla y aparece un error, si no, se inserta en la tabla y se muestra un mensaje de bienvenida
    if (mysqli_num_rows($resultado) > 0) {
        echo "<script>
            document.getElementById('regi').style.display = 'block';
            document.getElementById('regiError').innerHTML = 'El correo ya está registrado.';
        </script>";
    } else {
        $insertarUsu = "INSERT INTO Usuarios (nombre, apellidos, correo, contrasena)
            VALUES ('$nombre', '$ape', '$correo', '$contra')";

        if (mysqli_query($conn, $insertarUsu)) {
            echo "<script>
                alert('Bienvenido, $nombre');
                window.location.href = 'index.php?registroCorrecto=true';
            </script>";

            exit();
        } else {
            echo "Error: " . mysqli_error($conn);
        }
    }
    unset($_POST['registro']);
}
```

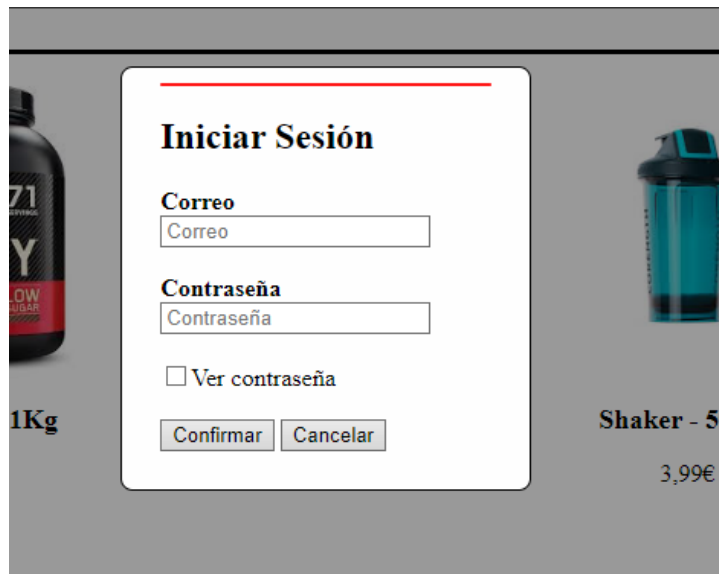
Primero, hacemos un if para que solo se ejecute este código cuando el método de envío sea post y que el botón que hemos pulsado para enviarlo tenga de nombre “registro”. Una vez hecho esto se recogen las variables introducidas y como podemos ver la contraseña se modifica con un hash para que en la tabla no aparezca explícitamente la contraseña introducida, en su lugar se mostrará algo como esto:

nombre	apellidos	correo	contrasena
Kevin	Nicolàs	ola1@ola1	\$2y\$10\$Z.TOFJqjD4pI9/ZizuXzh.vJW909WK4TFOXY1otROse...

Una vez recibidos los datos introducidos se comprueba que el correo no esté registrado ya en la tabla, pues este debería ser único para cada usuario. En caso de que esté registrado ya, se mostrará el mensaje que lo notifique, como se puede ver en una de las imágenes anteriores y, si no está registrado, se insertan los datos en la tabla, se muestra un mensaje de bienvenida, el cual está en una imagen anterior también y cambia la url con la función window.location.href, para que no suceda el error antes mencionado de que se vuelva a enviar el formulario al recargar la página.



## 4. Página de Inicio de Sesión



The image shows a login modal titled "Iniciar Sesión" (Login) overlaid on a product page. The modal contains two input fields: "Correo" (Email) and "Contraseña" (Password). Below the password field is a checkbox labeled "Ver contraseña" (Show password). At the bottom of the modal are two buttons: "Confirmar" (Confirm) and "Cancelar" (Cancel). The background shows a product page with a black bottle on the left and a blue shaker on the right. The bottle is labeled "71", "Y", and "LOW". The shaker is labeled "Shaker - 50" and "3,99€".

El funcionamiento del formulario de inicio de sesión en esta página es prácticamente igual al del registro, como se puede ver en la imagen, se utiliza otra vez un modal para poder ocultar el formulario en la misma página de bienvenida y que solo se muestre al pulsar el botón de iniciar sesión. La imagen de la siguiente página es del fichero que procesa los datos de este formulario.

```

// Verificar si el formulario de inicio de sesión fue enviado
if ($_SERVER["REQUEST_METHOD"] == "POST" && isset($_POST['iniciar'])) {

    // Obtener el correo y la contraseña del formulario
    $correo = $_POST['correo'];
    $contrasena = $_POST['contra'];

    // Consultar la base de datos para verificar si el correo está registrado
    $comprobarUsuario = "SELECT * FROM Usuarios WHERE correo = '$correo'";
    $resultado = mysqli_query($conn, $comprobarUsuario);

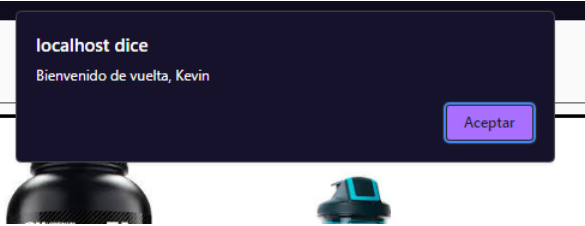

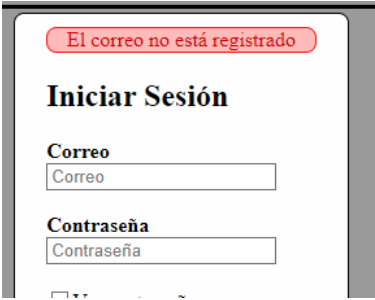
    // Si el resultado es mayor que 0, el usuario está en la tabla
    if (mysqli_num_rows($resultado) > 0) {
        $filaUsuario = mysqli_fetch_assoc($resultado);
        $contraBD = $filaUsuario['contrasena'];

        // Comparamos la contraseña ingresada con la almacenada en la base de datos
        if (password_verify($contrasena, $contraBD)) {
            //Si coinciden

            // Mostrar mensaje de bienvenida
            echo "<script>
                alert('Bienvenido de vuelta, {$filaUsuario['nombre']}');
                window.location.href = 'index.php?iniSes=true';
            </script>";
        } else {
            // Mostrar mensaje de contraseña incorrecta en el modal de inicio de sesión
            echo "<script>
                document.getElementById('ini').style.display = 'block';
                document.getElementById('iniError').innerHTML = 'Contraseña incorrecta';
            </script>";
        }
    } else {
        // Mostrar mensaje de correo no registrado en el modal de inicio de sesión
        echo "<script>
            document.getElementById('ini').style.display = 'block';
            document.getElementById('iniError').innerHTML = 'El correo no está registrado';
        </script>";
    }
    unset($_POST['iniciar']);
}

```

En este fichero podemos ver que, al igual que en el fichero que procesaba la información del formulario de registro, comprobamos que el método de envío sea post y que se haya pulsado el botón “iniciar”, si es así obtenemos el correo y la contraseña y comprobamos que haya una fila en la tabla con ese correo, si es así comprobamos que la contraseña introducida y la que hay en la tabla coincidan, si coinciden se muestra un mensaje de bienvenida. En caso de que no haya una fila con ese correo se mostrará un mensaje de error que dirá que el correo no está registrado, por otra parte, si el correo está registrado pero la contraseña no coincide con la de la tabla, se mostrará el error correspondiente.

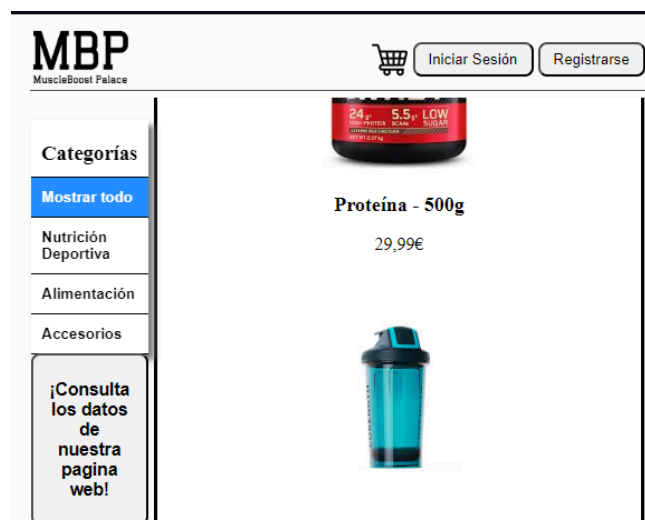
Posibles resultados del formulario de Inicio de Sesión	
	Inicio de sesión exitoso
	La contraseña no coincide con la que hay en la tabla usuarios
	El correo introducido no está en la tabla usuarios

## 5. Estilo

```
# estilos.css X
XAMPP-1 > # estilos.css > ...
1  body{
2      margin:0;
3      background-color: ■ rgb(250, 250, 250);
4  }
5
6  button{
7      cursor: pointer;
8  }
9
10 /*Contenido de la Cabecera*/
11
12 .cabecera{
13     padding: 1%;
14     margin-bottom: -4%;
15     margin-top: -1%;
16     background-color: ■ rgb(250, 250, 250);
17 }
18
19 .sticky {
20     position: fixed;
21     top: 0;
22     width: 98%;
23 }
24
25
26 .sticky + .contenido {
27     margin-top: 95px;
28 }
29
30 img {
31     max-width: 100%;
32     max-height: 100%;
33     object-fit: contain;
34 }
35
36 #divlogo{
37     max-width: 100%;
38     display: inline-block;
39     width: 90px;
40     margin-left: 2%;
41     overflow: auto;
42 }
43
```

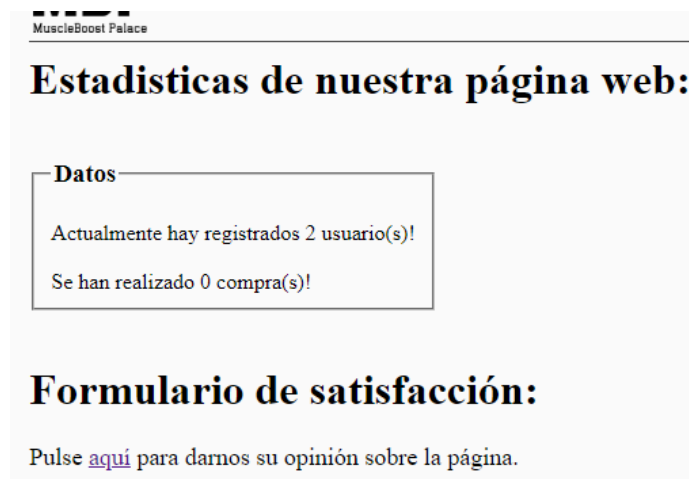
Como se ha podido ir apreciando en las imágenes que se han mostrado, la página utiliza un fichero css para hacer que sea más bonita visualmente, a la izquierda podemos ver un fragmento de las más de 270 líneas que forman parte de este archivo. Gracias a este fichero podemos hacer cosas como que solo se mueva la sección de los artículos cuando queremos desplazarnos arriba o abajo en la página, es decir, la cabecera y el apartado de categorías se quedarán fijos mientras movemos los artículos, esto no se puede apreciar muy bien en pantallas grandes, ya que se debería ampliar la cantidad de artículos para que fuera necesario desplazarse verticalmente en la web. Con este fichero también damos formato a los formularios, tipografía de la página, diseño de los botones y enlaces, configuramos que nuestra página sea responsive, es decir, que se vea bien en todo tipo de pantallas mediante porcentajes facilitando así que se cambie de tamaño automáticamente dependiendo de la anchura de la pantalla.

En la segunda imagen podemos apreciar que en una pantalla más pequeña los artículos se posicionan respecto a la anchura de la página, es decir, manteniendo el tamaño, se muestran los máximos artículos posibles por fila y los que no quepan, pasan a la siguiente fila, en esta imagen vemos un ejemplo de que solo se mueven los artículos, pues se ha desplazado verticalmente la página y los artículos se esconden detrás del header, las categorías tampoco se han movido y además han cambiado su tamaño para poder caber mejor en la página, todo esto sin perder su funcionalidad. Todas estas implementaciones hacen que esta página se vaya a ver bien tanto en un



ordenador de sobremesa como en un dispositivo móvil.

## 6. Página de consultas



Esta es la información que podemos encontrar en la página de consultas, como se puede ver en la imagen, tenemos un título realizado con la etiqueta h2, un fieldset con su respectivo legend para agrupar los datos que se muestran. En el código, utilizamos php para realizar la consulta, entonces, para que se pueda ver el resultado de la consulta (el número de usuarios y de compras) dentro de una frase, he optado por utilizar php inmediatamente después del título, es decir, el fieldset se realiza dentro del php mediante un echo, como se puede ver en la imagen inferior. Para obtener el resultado de los usuarios y las compras se realiza una consulta a la base de datos que cuenta cuántas filas hay en cada tabla respectivamente y el número mostrado es el resultado de esta consulta.

```
//Consulta el numero de filas de la tabla usuarios
$consulUsu = "SELECT count(correo) as totalUsuarios FROM Usuarios";
$resultUsu = mysqli_query($conn, $consulUsu);

// Verifica si la consulta fue exitosa
if ($resultUsu) {
    $row = mysqli_fetch_assoc($resultUsu);
    $totalUsuarios = $row['totalUsuarios'];
    echo "<fieldset style='display: inline;*><legend*>Datos</legend*>Actualmente hay registrados " . $totalUsuarios . " usuario(s)!";
} else {
    echo "Error al ejecutar la consulta: " . mysqli_error($conn);
}

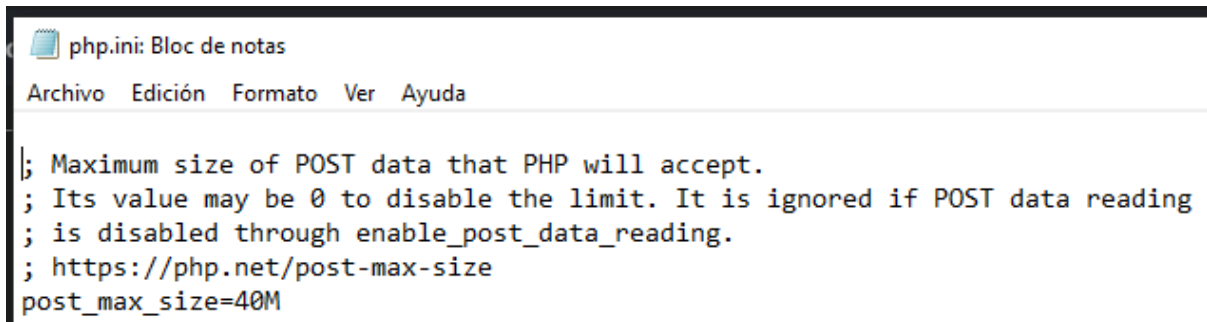
//Consulta el numero de filas de la tabla compras
$consulComp = "SELECT count(id_compra) as totalCompras FROM Compras";
$resultComp = mysqli_query($conn, $consulComp);

// Verifica si la consulta fue exitosa
if ($resultComp) {
    $row = mysqli_fetch_assoc($resultComp);
    $totalCompras = $row['totalCompras'];
    echo "<br*><br*>Se han realizado " . $totalCompras . " compra(s)!</fieldset*>";
} else {
    echo "Error al ejecutar la consulta: " . mysqli_error($conn);
}
```

Como podemos ver en la imagen, tenemos las consultas para cada tabla, el fieldset que engloba ambos resultados y una concatenación en cada echo para poder mostrar correctamente el resultado de la consulta.

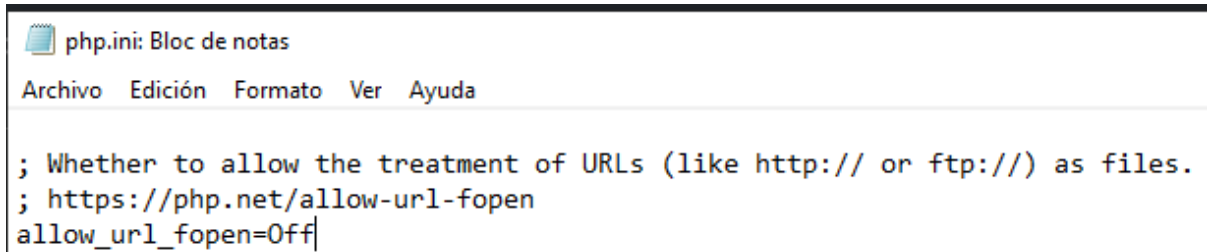
En la misma página de la consulta hay un enlace a un formulario realizado con Google Forms para que el usuario pueda dar su opinión sobre la página, se profundizará en esto más adelante.

## 7. Añadir medidas de seguridad



```
php.ini: Bloc de notas
Archivo Edición Formato Ver Ayuda

; Maximum size of POST data that PHP will accept.
; Its value may be 0 to disable the limit. It is ignored if POST data reading
; is disabled through enable_post_data_reading.
; https://php.net/post-max-size
post_max_size=40M
```



```
php.ini: Bloc de notas
Archivo Edición Formato Ver Ayuda

; Whether to allow the treatment of URLs (like http:// or ftp://) as files.
; https://php.net/allow-url-fopen
allow_url_fopen=Off
```

Tal y como muestran las dos capturas superiores, se ha editado el fichero `php.ini` para que se limite el tamaño de los ficheros a 40Mb y se ha establecido la directiva `allow_url_fopen` a off. Con estos cambios en la configuración se apunta a mejorar la seguridad de la página web.

## 8. Extra

### 8.1. Carrito

**MBP**  
MuscleBoost Palace

Volver

### Este es su carrito de la compra

Escoja los productos que desee comprar

Pulse para añadir más productos

Total 63.97€

Comprar

Eliminar Elija un producto Proteina - 500g Cantidad: 2 Precio: 59.98€

Eliminar Elija un producto Shaker - 500ml Cantidad: 1 Precio: 3.99€

La página incorpora una sección de carrito en la cual el usuario puede añadir los productos que desea comprar y la cantidad, entonces se le mostrará el precio de esa línea (precio del producto por la cantidad seleccionada) en tiempo real, además del total de la compra (suma del precio de todas las líneas). Se pueden añadir tantas líneas como se desee, también se pueden quitar a elección una línea o varias, incluso todas las líneas si se quiere, pues el formulario a partir del cual se clonan estas líneas está oculto y no se podrá borrar nunca por accidente. Al pulsar el botón de comprar, todos los formularios (líneas) que haya en la página serán enviados a un php que los procesará todos simultáneamente, se mostrará un mensaje de confirmación y los productos que hubiera seleccionados, junto con su cantidad y su precio total, se insertarán en la tabla “Compras”.

**MBP**  
MuscleBoost Palace

Volver

## Gracias por su compra, vuelva pronto

Volver a la página principal

Desde esta página podremos volver a la página principal de tres formas distintas:

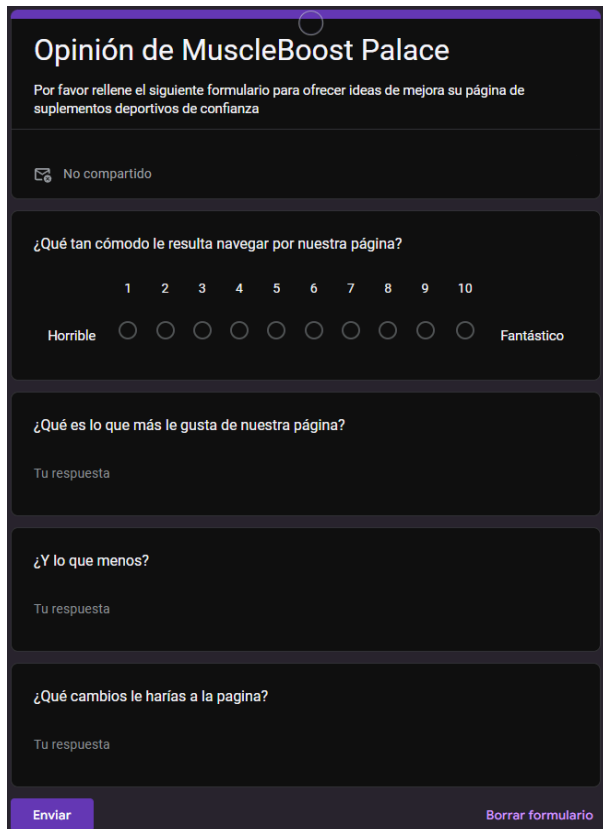
- 1 - Pulsando el botón “Volver a la página principal”
- 2 - Pulsando el botón “Volver”
- 3 - Pulsando en el logo de la página

id_compra	producto	cantidad	precioTot
35	Proteina - 500g	2	59.98
36	Shaker - 500ml	1	3.99

Este es el resultado que obtendremos en la tabla, como se puede ver se han almacenado correctamente ambos productos, además de su cantidad y precio

total. Hay una columna “id\_compra” que es un contador, es decir, cada vez que se añade una línea en esta tabla, su valor se incrementa automáticamente, eso se hace así para que no haya dos productos con el mismo id, pues es la clave primaria de esta tabla, permitiendo con ello que se puedan encontrar los productos comprados más fácilmente.

## 8.2. Aplicaciones de Ofimática Web



The screenshot shows a Google Form titled "Opinión de MuscleBoost Palace". The subtitle reads: "Por favor rellene el siguiente formulario para ofrecer ideas de mejora su página de suplementos deportivos de confianza". Below the title, there is a status bar indicating "No compartido". The form contains four questions:

- Question 1: "¿Qué tan cómodo le resulta navegar por nuestra página?". It features a horizontal scale from 1 to 10, with "Horrible" at the left end and "Fantástico" at the right end. The scale is currently empty.
- Question 2: "¿Qué es lo que más le gusta de nuestra página?". It has a text input field labeled "Tu respuesta".
- Question 3: "¿Y lo que menos?". It also has a text input field labeled "Tu respuesta".
- Question 4: "¿Qué cambios le harías a la página?". It has a text input field labeled "Tu respuesta".

At the bottom of the form, there are two buttons: "Enviar" (green) and "Borrar formulario" (red).

Como se ha mencionado anteriormente, en la página se ha incorporado un formulario realizado con Google Forms para conocer la opinión de los usuarios respecto al uso y comodidad de la web. Como se puede apreciar en la imagen, el formulario consta de cuatro preguntas las cuales servirán para saber qué piensan los clientes respecto a la comodidad, cuáles creen que son los puntos fuertes, cuáles los débiles y qué harían para mejorar la página. Además, para implementar el uso de la ofimática web, este documento ha sido realizado con la ayuda de Google Docs.

## 8.3. Información Adicional

Junto con este documento se adjuntan los ficheros de la página (index.php, que es la página principal, scripts.php, que es el archivo que crea la base de datos y las tablas, etc.) para poder ver de primera mano todo lo que se ha realizado. Los archivos tienen incorporados comentarios para que se pueda entender lo que hace cada apartado, en caso de que algún apartado no esté explicado correctamente o deje dudas sobre lo que realiza, el diseñador de la web asume la responsabilidad de realizar la aclaración verbalmente.

Para lograr la correcta ejecución de los ficheros, si se quiere simular el mismo entorno en el cual han sido concebidos, se debería utilizar XAMPP y phpMyAdmin (incorporado al utilizar XAMPP).