

Indice

1. Objetivos.
2. Plan de empresa.
 - 2.1 Presentación de la empresa
 - 2.2 Presentación de los promotores
 - 2.3 Análisis de la idea de negocio
 - 2.4 Plan de marketing
 - 2.5 Plan de producción y calidad
 - 2.6 Plan de organización
3. Página web
 - 3.1 HTML
 - 3.2 CSS
 - 3.3 PHP
 - 3.4 Conectar BBDD
 - 3.5 Forma de compra
 - 3.6 Usuarios
 - 3.7 Estructura Web
 - 3.8 Carrito de la compra
4. Primeras impresiones y problemas
5. Pago

1. Objetivos

En este proyecto se intenta aunar los conocimientos visto durante el curso de la asignatura: implantación de aplicaciones web y empresa e iniciativa emprendedora. Para ello el proyecto consiste en la representación de una empresa ficticia.

Para la parte de implantación de aplicaciones web se realizara una página web para dicha empresa, donde se reunirán los conocimientos de recopilados durante el curso.

Para la parte de empresa e iniciativa emprendedora se creara un plan de empresa como el que se podría hacer para crear una empresa real.

2. Plan de empresa

Definición:

Es un documento en que se define de forma clara y sintética la idea de negocio que se desea emprender. Detalla el contenido del proyecto empresarial. Abarca desde la definición de la idea a desarrollar hasta la forma concreta de llevarla a la práctica, incluyendo la programación de las fases, medios y recursos a utilizar.

El plan de empresa tiene dos finalidades principales:

- Obliga al emprendedor a hacer un esfuerzo de organización, planificación y eficacia previa a la puesta en marcha del proyecto, que le permite evaluarlo y determinar la viabilidad de la idea.
- Es un carta de presentación del emprendedor frente a terceros (clientes, posibles inversores, instituciones financieras, organismos públicos, etc.)

Datos a tener en cuenta:

- Tiene que basarse en datos recientes.
- Debe contemplar aspectos relevantes para la empresa.
- Debe ser claro y comprensible.

2.1 Presentación de la empresa

En la presentación de la empresa se debe explicar cuál es la idea de negocio que se quiere llevar a cabo. De forma fluida y ordenada tenemos que explicar lo básico de nuestra empresa, como se llama, a que se va a dedicar, quienes van a ser los clientes y donde se va a situar. En pocas palabras es un resumen de la empresa.

Conviene que siga una estructura:

- Nombre de la empresa
- Localización
- Descripción de la idea de negocio (¿Cuál será la actividad de la empresa?)
- Necesidades que satisface
- Mercado: Clientes
- Política de Calidad

2.2 Presentacion de los promotores

En este punto hay que presentar a los promotores.

Explicación de quién es el empresario o los empresarios su experiencia empresarial, conocimientos del mercado o actividad.

Grado de vinculación al negocio (% capital, trabajo,..).

2.3 Análisis de la idea de negocio

Para realizar el análisis de la idea de negocio se utiliza el método DAFO

Debilidades: son los puntos débiles de la empresa, que limitan o reducen la capacidad de desarrollo de su estrategia y deben ser controladas y superadas.

Amenazas: toda fuerza del entorno que puede impedir la implantación de una estrategia, reducir su efectividad o los recursos que se requieren para su implantación.

Fortalezas: son los puntos fuertes de la empresa, sus capacidades, recursos, posiciones alcanzadas, etc.

Oportunidades: todo aquello que pueda suponer una ventaja competitiva para la empresa, o posibilidad de mejorar la rentabilidad de la misma o aumentar su cifra de negocios.

Con la ayuda de los resultados del análisis DAFO la empresa podrá hacerse una idea de que estrategia tomar:

- **Defensiva:** La empresa debe estar preparada para defenderse de las amenazas. Si el producto o servicio ya no es líder tendrá que resaltar lo que le diferencia de la competencia. Cuando nos baje la cuota de mercado buscaremos clientes rentables y protegerlos.
- **Ofensiva:** La empresa debe adoptar una estrategia de crecimiento. Cuando las fortalezas son reconocidas por sus clientes, es posible atacar a la competencia para resaltar las ventajas propias. Cuando el mercado está maduro es posible tratar de “robar” clientes lanzando nuevos modelos o servicios.
- **Supervivencia:** La empresa se enfrenta a amenazas externas sin las fuerzas internas necesarias para luchar contra la competencia. Es aconsejable dejar las cosas tal y como están hasta que se asienten los cambios que se producen.

- Reorientación: Aparecen oportunidades que se pueden aprovechar, pero la empresa carece de la preparación adecuada. Será necesario cambiar de política o de producto o servicio porque los actuales no están dando los resultados esperados.

2.4 Plan de marketing

Estudio del sector en el que la empresa se va a mover para determinar la cuota de mercado inicial.

En consecuencia, idearemos un plan de *marketing estratégico-operativo* que distinga a nuestra empresa de otras del sector y la haga más atractiva a potenciales compradores. Determinaremos la estrategia de precios, política de ventas (distribuidores, comerciantes, plazos de pago, promociones, descuentos,...) publicidad (en radio local, buzoneo, mailing, revistas del sector) servicio postventa y garantía.

2.5 Plan de producción y calidad

Es una descripción de la realización del producto o prestación del servicio (explicación del proceso de producción):

- Proveedores con los que se va a trabajar, recepción materias primas, elaboración de producto o servicio, almacenaje, comercialización, distribución).
- Control de calidad, tecnología utilizada.
- Instalaciones y maquinaria utilizada

2.6 Plan de organización

Explica cómo se va a organizar la empresa, para que sea fácil de comprender se realiza a través de un organigrama. Donde se explica en que departamentos se va a dividir la empresa, así como la función que va a realizar cada departamento, el número de trabajadores y el perfil exigido en cada puesto de trabajo.

3. Pagina web

Una página web es un documento o información electrónica capaz de contener texto, sonido, vídeo, programas, enlaces, imágenes, y muchas otras cosas y que puede ser accedida mediante un navegador. Esta información se encuentra generalmente en formato HTML o XHTML, y puede proporcionar navegación (acceso) a otras páginas web mediante enlaces de hipertexto. Las páginas web frecuentemente también incluyen otros recursos como ser hojas de estilo en cascada, guiones (scripts), imágenes digitales, entre otros.

Las páginas web pueden estar almacenadas en un equipo local o un servidor web remoto. El servidor web puede restringir el acceso únicamente a redes privadas, por ejemplo, en una intranet corporativa, o puede publicar las páginas en la World Wide Web. El acceso a las páginas web es realizado mediante una transferencia desde servidores, utilizando el protocolo de transferencia de hipertexto (HTTP).

Una página web está compuesta principalmente por información (sólo texto y/o módulos multimedia) así como por hiperenlaces, además puede contener o asociar hoja de estilo, datos de estilo para especificar cómo debe visualizarse, y también aplicaciones para así permitir interactividad.

Las páginas web son escritas en un lenguaje de marcado que provee la capacidad de manejar e insertar hiperenlaces, generalmente HTML.

El contenido de la página puede ser predeterminado (página web estática) o generado al momento de visualizarla o solicitarla a un servidor web (página web dinámica). Las páginas dinámicas que se generan al momento de la visualización, se especifican a través de algún lenguaje interpretado, generalmente JavaScript, y la aplicación encargada de visualizar el contenido es la que realmente debe generarlo. Las páginas dinámicas que se generan, al ser solicitadas, son creadas por una aplicación en el servidor web que alberga las mismas.

Respecto a la estructura de las páginas web, algunos organismos, en especial el World Wide Web Consortium (W3C), suelen establecer directivas con la intención de normalizar el diseño, y para así facilitar y simplificar la visualización e interpretación del contenido.

Una página web es en esencia una tarjeta de presentación digital, ya sea para empresas, organizaciones, o personas, así como una tarjeta de presentación de ideas y de informaciones y de teorías

3.1 HTML

Siglas de HyperText Markup Language («lenguaje de marcas de hipertexto»), hace referencia al lenguaje de marcado para la elaboración de páginas web. Es un estándar que sirve de referencia para la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes, etc. Es un estándar a cargo de la W3C, organización dedicada a la estandarización de casi todas las tecnologías ligadas a la web, sobre todo en lo referente a su escritura e interpretación.

El lenguaje HTML basa su filosofía de desarrollo en la referenciación. Para añadir un elemento externo a la página (imagen, vídeo, script, etc.), este no se incrusta directamente en el código de la página, sino que se hace una referencia a la ubicación de dicho elemento mediante texto. La página web contiene sólo texto mientras que recae en el navegador web la tarea de unir todos los elementos y visualizar la página final. Al ser un estándar, HTML busca ser un lenguaje que permita que cualquier página web escrita en una determinada versión, pueda ser interpretada de la misma forma por cualquier navegador web actualizado.

Sin embargo, a lo largo de sus diferentes versiones, se han incorporado y suprimido diversas características, con el fin de hacerlo más eficiente y facilitar el desarrollo de páginas web compatibles con distintos navegadores y plataformas (PC, portátiles, teléfonos móviles, tabletas, etc.).

El lenguaje HTML puede ser creado y editado con cualquier editor de textos básico, como puede ser Gedit en Linux, el Bloc de notas de Windows, o cualquier otro editor que admita texto sin formato como GNU Emacs, Microsoft Wordpad, TextPad, Vim, Notepad++, entre otros.

Existen, además, otros editores para la realización de sitios web con características WYSIWYG (What You See Is What You Get, o en español: «lo que ves es lo que obtienes»). Estos editores permiten ver el resultado de lo que se está editando en tiempo real, a medida que se va desarrollando el documento. Estos programas, además de tener la opción de trabajar con la vista preliminar, tiene su propia sección HTML, la cual va generando todo el código a medida que se va trabajando. Algunos ejemplos de editores WYSIWYG son KompoZer, Microsoft FrontPage o Adobe Dreamweaver.

HTML utiliza etiquetas o marcas, que consisten en breves instrucciones de comienzo y final, mediante las cuales se determina la forma en la que debe aparecer en su navegador el texto, así como también las imágenes y los demás elementos, en la pantalla del ordenador.

```
<html>
  <head>
    <title>Ejemplo1</title>
```

```
</head>
<body>
  <p>ejemplo1</p>
</body>
</html>
```

3.2 CSS

CSS o Cascading Style Sheets (Hojas de Estilo en Cascada) es el lenguaje utilizado para describir el aspecto y el formato de un documento escrito en un lenguaje de marcas, esto incluye varios lenguajes basados en XML como son XHTML o SVG.

La información de estilo puede ser adjuntada como un documento separado o en el mismo documento HTML. En este último caso podrían definirse estilos generales en la cabecera del documento o en cada etiqueta particular mediante el atributo "<style>".

Las especificaciones de CSS son mantenidos por el Consorcio World Wide Web (W3C). Tipo de medio de Internet (tipo MIME) text/css está registrado para su uso con CSS porRFC 2318 (marzo de 1998), y también opera un libre servicio de validación de CSS .

Por ejemplo, el elemento de HTML <h1> indica que un bloque de texto es un encabezamiento y que es más importante que un bloque etiquetado como <h2>. Cuando se utiliza CSS, la etiqueta <h1> no debería proporcionar información sobre cómo será visualizado, solamente marca la estructura del documento. La información de estilo, separada en una hoja de estilo, especifica cómo se ha de mostrar <h1>: color, fuente, alineación del texto, tamaño y otras características no visuales.

Antes de que estuviera disponible CSS, la única forma de componer espacialmente una página era el uso de tablas <table>. Aunque esta era una técnica cómoda y versátil, ello conllevaba el uso de un elemento con una semántica particular, y en el que la distribución de los datos no se ajustaban al flujo de la información que se obtenía en la vista desde los navegadores habituales.

Mediante el uso de CSS, se ha permitido eliminar el uso de tablas para el diseño, usándolas solamente para la muestra de datos tabulados.

Algunas ventajas de utilizar CSS son:

- Control centralizado de la presentación de un sitio web completo con lo que se agiliza de forma considerable la actualización del mismo.
- Separación del contenido de la presentación, lo que facilita al creador, diseñador, usuario o dispositivo electrónico que muestre la página, la modificación de la visualización del documento sin alterar el contenido del mismo, sólo modificando algunos parámetros del CSS.

- Optimización del ancho de banda de la conexión, pues pueden definirse los mismos estilos para muchos elementos con un sólo selector; o porque un mismo archivo CSS puede servir para una multitud de documentos.
- Mejora en la accesibilidad del documento por parte de navegadores orientados a personas con algunas limitaciones sensoriales.

```
<html>

  <head>

    <title>Mi primera página con estilo</title>

    <style type="text/css">

      body {

        color: purple;

        background-color: #d8da3d }

    </style>

  </head>

  <body>
```

3.3PHP

PHP es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página Web resultante. PHP ha evolucionado por lo que ahora incluye también una interfaz de línea de comandos que puede ser usada en aplicaciones gráficas independientes.

Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. Es también el módulo Apache más popular entre las computadoras que utilizan Apache como servidor web.

El gran parecido que posee PHP con los lenguajes más comunes de programación estructurada, como C y Perl, permiten a la mayoría de los programadores crear aplicaciones complejas con una curva de aprendizaje muy corta. También les permite

involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones.

Aunque todo en su diseño está orientado a facilitar la creación de sitios webs, es posible crear aplicaciones con una interfaz gráfica para el usuario, utilizando alguna extensión como puede ser PHP-Qt, PHP-GTK,⁶ WxPHP, WinBinder, Roadsend PHP, Phalanger, Phc o HiP Hop VM.

Cuando el cliente hace una petición al servidor para que le envíe una página web, el servidor ejecuta el intérprete de PHP. Éste procesa el script solicitado que generará el contenido de manera dinámica (por ejemplo obteniendo información de una base de datos). El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente.

Mediante extensiones es también posible la generación de archivos PDF,⁸ Flash, así como imágenes en diferentes formatos.

Permite la conexión a diferentes tipos de servidores de bases de datos tanto SQL como NoSQL tales como MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird, SQLite o MongoDB.⁹

PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos, tales como Unix (y de ese tipo, como Linux o Mac OS X) y Microsoft Windows, y puede interactuar con los servidores de web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI.

PHP es una alternativa a las tecnologías de Microsoft ASP y ASP.NET (que utiliza C# y Visual Basic .NET como lenguajes), a ColdFusion de la empresa Adobe, a JSP/Java, CGI/Perl y a Node.js/Javascript. Aunque su creación y desarrollo se da en el ámbito de los sistemas libres, bajo la licencia GNU, existe además un entorno de desarrollo integrado comercial llamado Zend Studio. CodeGear (la división de lenguajes de programación de Borland) ha sacado al mercado un entorno de desarrollo integrado para PHP, denominado 'Delphi for PHP'. También existen al menos un par de módulos para Eclipse, uno de los entornos más populares.

```
<html lang="es">
  <head>
    <meta charset="UTF-8" />
    <title> Ejemplo básico PHP</title>
  </head>
```

```
<body>
  <?php
    echo 'Hola mundo';
  ?>
</body>
</html>
```

3.4 Conectar base de datos

La conexión de la base de datos con la página se realiza a través de código php.

Para realizar la conexión se utiliza una variable que tenga el siguiente valor (`mysql_connect("servidor," "usuario," "contraseña");`), sustituyendo “servidor” por el nombre o la dirección ip del servidor donde se encuentra alojada la base de datos, “usuario” lo sustituimos por el usuario que queremos acceder a la base de datos, la forma más fácil es poniendo el usuario administrador pero esto sería un gran fallo de seguridad por lo que sería mejor crear un usuario con menos permisos o incluso ir cambiando de usuarios con los permisos limitados para realizar solo lo que queremos, “contraseña” este campo indica la contraseña del usuario anterior nombrado.

Después de esta línea debemos indicar a que base de datos queremos acceder utilizando el siguiente comando (`mysql_select_db("base de datos");`) sustituimos “base de datos” por el nombre de la base de datos a la que queremos acceder.

Esta operación se tiende a realizar una vez después de la autenticación del usuario que accede a la página web, una vez hecho esta información queda almacenada hasta que el usuario cierre la sesión actual. Todo este proceso se hace oculto al usuario, el solo vera como accede con su cuenta a la página pero no como la página ha accedido a la base de datos.

3.5 Forma de compra

Sopesamos los diferentes modelos de compra, Tarjeta, transferencia, paypal etc etc,

Con tarjeta habría que hablar con el banco para que nos verifica la tarjeta de crédito del cliente y nos haga un túnel seguro con certificado, nosotros tendríamos que enviar el importe a la página del banco y recibir si ha dado o no el ok, en un principio lo descartamos porque aparte de que el banco se llevaría una jugosa comisión, y para empezar no queremos que sea así, por paypal habría que habilitar el modo seguro y enviar el importe del pedido a una url que redireccione a nuestra cuenta para que paguen y recoger si ha dado trae o false el pago.

Al final nos decantamos por lo más sencillo, la transferencia bancaria, enviando un e-mail al cliente con el importe total y los productos comprados a través de un servidor de correo y con el número de cuenta y el número de pedido para que lo ponga como asunto en el pago

3.6 Usuarios

Los Usuarios los haremos de dos tipos, uno administrador creado a mano y los demás que se creen como usuario, redireccionando a un sitio a otro de la siguiente manera:

```
If isset ($sesion){  
    If ($sesion='Administrador')  
        {include formulario/menu de administrador en la parte superior}  
    Else  
        {include formulario/menu de usuario en la parte superior}  
    }  
else {  
    incluye(iniciar_sesion.php)  
}
```

las capas se incluirán usando include, para que aunque el usuario no esté logueado muestre los productos

Para el logueo en la base de datos usaremos una de las prácticas echas en clase según lo expuesto antes.

3.7 Estructura página web

Con el fin de que la página sea modulable y la podamos modificar fácilmente hemos decidido crear una estructura por capas, así podemos dejar siempre visibles unas capas con la información que le pasemos y será más fácil la instalación de plugins en jquery por ejemplo.

```
<html>  
<head>  
<head><title>Principal</title></head>  
<body>  
<div id="logo">  
</div>  
<div id="botones">  
</div>
```

```

<?php
If isset ($sesion){
    If ($sesion=' Administrador')
        {include formulario/menu de administrador en la parte superior}
    else
        {include formulario/menu de usuario en la parte superior}
}
?>

<div id="inicio">

</div>

<div id="principal">

</div>

<div id="carrito">

</div>

</body></html>

```

3.8 Carrito de la compra

Para realizar las compras utilizamos un plugin que hace la función de carrito de la compra.

Para que el plugin funcione hay que programarlo correctamente. Debemos crear una lista con los artículos que tenemos en la página y su información. Modificando el archivo config.php del plugin , de la siguiente manera:

```

<?php
$config['item']['id']      = 'my-item-id'; // Item id
$config['item']['name']   = 'my-item-name'; // Item name
$config['item']['price']  = 'my-item-price'; // Item price
$config['item']['qty']    = 'my-item-qty'; // Item quantity
$config['item']['url']    = 'my-item-url'; // Item URL (optional)
$config['item']['add']    = 'my-add-button'; // Add to cart button
?>

```

Hay que asegurarse de que los artículos de la página también tienen esos atributos:

También hay que asegurar se de incluir la siguiente línea en cada página.

```

<form method="post" action="jcart/jcart.php">
  <fieldset>
    <input type="hidden" name="my-item-id" value="1" />
    <input type="hidden" name="my-item-name" value="Soccer Ball" />
    <input type="hidden" name="my-item-price" value="25.00" />
    <input type="hidden" name="my-item-url" value="" />

    <ul>
      <li><strong>Soccer Ball</strong></li>
      <li>Price: $25.00</li>
      <li>
        <label>Qty: <input type="text" name="my-item-qty" value="1" size="3" /></label>
      </li>
    </ul>

    <input type="submit" name="my-add-button" value="add to cart" class="button" />
  </fieldset>
</form>

include_once ( 'jcart / jcart.php' );

```

Para que se muestre la cesta de la compra debemos incluir el siguiente código:

```
<php $jcart? -> display_cart (); ?>
```

Por ultimo hay que asegurarse de añadir class = "jcart" a los formularios en los que aparecen los artículos para utilizar el plugin.

En la capa carrito lo pondremos:

```
<div id="jcart"><?php $jcart->display_cart();?></div>
```

Por lo tanto el id de la capa carrito es cambiado a jcart y en el head añadimos lo siguiente:

```

<script type="text/javascript" src="jcart/js/jquery 1.4.4.min.js"></script>
<script type="text/javascript" src="jcart/js/jcart.min.js"></script>

```

Con esto ya tenemos el carrito configurado, procederemos a hacer la web.

Para enviar un correo procederemos a usar la función mail de PHP

```

<?PHP

foreach ($cart->get_contents() as $item)

```

```

{
    $item_id = $item['id'];
    $item_name = $item['name'];
    $item_price = $item['price'];
    $item_qty = $item['qty'];

    $mensaje.= $item_id."-".$item_name."-".$item_price."$"."-".$item_qty."<br>";
    echo $item_id."-";
    echo $item_name."-";
    echo "$".$item_price."-";
    echo $item_qty;
    echo "<br>";
    mail('direccion@dominio.extension', 'titulo', $mensaje);

?>

```

La capa de carrito quedaría así:

```

include("conectar.php");

$registros = 15;
$pagina = $_GET["pagina"];

if (!$pagina) {
    $inicio = 0;
    $pagina = 1;
}
else {
    $inicio = ($pagina - 1) * $registros;
}

$resultados = mysql_query("SELECT id FROM productos WHERE visible = 1");
$total_registros = mysql_num_rows($resultados);
$resultados = mysql_query("SELECT * FROM productos WHERE visible = 1 LIMIT $inicio, $registros");
$total_paginas = ceil($total_registros / $registros);

$d=1;

// Primero, debemos ver nuestros elementos...

echo '<table border="0" cellpadding="0" cellspacing="15" width="607">';

```



```

$contador =1;

$num_col=4;

echo "<tr>";

while ($row=mysql_fetch_assoc($resultados)){

    echo "<td>";

    echo "<form method='post' action="" class='jcart'> ";

    echo "<input type='hidden' name='my-item-id' value='$d' />";
    echo "<input type='hidden' name='my-item-name' value='$row[titulo]' />";
    echo "<input type='hidden' name='my-item-price' value='$row[precio]' />";
    echo "<b><center>$row[titulo]</b><br /><img src='$row[imagen]'width='150' height='150' class='bor' /><br /><br><img
size:15px>$row[precio]</span><br> <img src='images/icon.gif' /> Cantidad: <input type='text' name='my-item-qty' value='1

$d++;

if($_SESSION['id'])
{ echo "<center><input type='submit' name='my-add-button' value='.' class='button' /></center>";

    }
else
{
    echo "<center><span style='color:red;border:1px solid red;padding:4px'>Inicie Sesión</span></center>";
}

echo "<div class='dash3'></div>";

echo " " </form>";

    echo "</td>";

    if ($contador == $num_col){

        echo "</tr><tr>";

        $contador=0;

    }

    $contador++;

}

```

```

//terminar de rellenar tablacon celdas...

if ($contador < $num_col){

    $celdas_falt=$num_col - $contador;

    while ($celdas_falt <= $num_col){

        echo "<td>$nbsp</td>";

        $celdas_falt++;

    }

    echo "</tr>";

}

echo '</table>';

mysql_free_result($resultados);

if($total_registros) {

    echo "<center>";

    if(($pagina - 1) > 0) {
        echo "<a href='pedido2.php?pagina=" . ($pagina-1) . "' style='border: 1px solid #ccc;padding: 3px;'>< Anterior</a> ";
    }

    for ($i=1; $i<=$total_paginas; $i++){
        if ($pagina == $i) {
            echo "<b style='border: 1px solid #ccc;padding: 3px;'>".$pagina."</b> ";
        } else {
            echo "<a href='pedido2.php?pagina=$i' style='border: 1px solid #ccc;padding: 3px;'>$i</a> ";
        }
    }

    if(($pagina + 1)<=$total_paginas) {
        echo " <a href='pedido2.php?pagina=" . ($pagina+1) . "' style='border: 1px solid #ccc;padding: 3px;'>Siguiete ></a>";
    }

}

```

```

    echo "</center>" ;
?>

</div>

Muy importante importar el carrito antes que el sesion_start:

// INCLUDE JCART BEFORE SESSION START
include 'jcart-1.1/jcart-1.1/jcart/jcart.php';

// START SESSION
session_start();

// INITIALIZE JCART AFTER SESSION START

$cart =& $_SESSION['jcart']; if(!is_object($cart)) $cart = new jcart();

?>

```

4. Primeras impresiones y problemas

Con esto la página quedaría así:





Para categorías en la base de datos creamos un campo en la BD tipo que nos dice si es una placa base o no, certificado con un check, también podemos ver en la parte de arriba una capa, dónde si hacemos clic en Su cuenta nos lleva a un formulario de registro:



Para que sólo cambie la capa principal hemos hecho una página web authentication.php con include par ano meter todo de nuevo y en la principal se muestre el formulario de acceso, por supuesto todas las capas con position relative en CSS.

Ya en mi cuenta hemos hecho un menú con diferentes consultas:

[Inicio](#) > [Mi cuenta](#)**MI CUENTA****Bienvenido a su cuenta. Desde aquí puede administrar las direcciones y los pedidos.** [Historia y detalles de mis pedidos](#) [Mis crédito](#) [Mis direcciones](#) [Mis datos personales](#) [Mis vales descuento](#) [Inicio](#)

Un problema ha sido historia y detalles de mis pedidos, hemos creado la tabla pedido siendo la clave primaria tando el id del pedido cómo el id_artículo y la consulta es un select distinct id from pedidos where usuario = '\$user'. Y que cada pedido sea un link a una consulta que se pase el id del pedido y ya saque los artículos y la cantidad, lo hemos hecho con un formulario, y la consulta es select a.nombre, a.tipo, p.cantidad from artículos a, pedidos p where '\$user' = p.usuario and a.id = p.id_articulo and p.id = '\$pid' y a la derecha el valor total con un select sum(p.precio) from artículos a, pedidos p where '\$user' = p.usuario and a.id = p.id_articulo and p.id = '\$pid'

Hemos tenido que añadir el campo usuario y precio a la tabla de pedidos y calcular el valor total del pedido para que a la hora de añadir el producto, que multiplique el precio por la cantidad y esa cantidad vaya a la base de datos y así pueda mostrar el precio total con un sum.


La opción de vales descuento no funciona y las demás son simples selects.

5. Pago


[Resumen](#) [Login](#) [Dirección](#) [Envío](#) [Pago](#)

● ○ ○ ○ ○

Último producto añadido


Gigabyte GA-Z87X-UD3H

Su carrito contiene 1 producto

Producto	Descripción	Ref.	Dispo.	Precio unitario	Cantidad	Total
	Gigabyte GA-Z87X-UD3H	--	✓	139,00 €	<div style="display: flex; align-items: center;"> + <input style="width: 40px; text-align: center;" type="text" value="1"/> - </div>	139,00 €
Total de productos IVA incluido:						139,00 €
Total sin IVA:						139,00 €
Total de impuestos:						0,00 €
Total IVA incluido:						139,00 €

Vales descuento Código: AÑADIR

En esta sección sólo tenemos que incluir el plugin y pasarle los campos como está comentado más arriba.

Al acabar la compra saldrá este mensaje (los datos están vacío porque la variable \$cuenta está vacía) y se enviará un e-mail al cliente como a mí a través de la función mail de PHP, lo malo que llega a SPAM...:

Su pedido **WorldPC** está completo.

por favor, háganos una transferencia por:

- un importe de **139,00 €**

- al propietario de la cuenta _____

- con los siguientes datos _____

- a este banco _____

- No se olvide añadir su pedido # **2** en el objeto de su transferencia

Le hemos enviado un email con esta información.

Su pedido será enviado próximamente, en cuanto recibamos su pago.

Para cualquier duda, por favor contacte con nosotros [atención al cliente](#).



[Volver a pedidos](#)