

## TEMA 4

### DESARROLLO DE UNA APLICACIÓN COMPLETA EN PHP

#### INTRODUCCIÓN

El objetivo de este tema es poner en práctica todo lo aprendido hasta el momento en los apartados estudiados.

Pretendemos realizar un aplicación web completa para realizar pedidos, similar a una tienda *on line*

Se tratará de un pequeño proyecto, pero deberá integrar todas la fases que se siguen en un proyecto real, Análisis, Diseño, Implementación, Mantenimiento, Seguimiento, así como diseño y desarrollo de la base de datos, y pruebas de ejecución y de carga de datos.

La elección del tipo de aplicación es **libre, así como el alcance** y se deja a criterio del alumno, pero tiene que hacer uso de los lenguajes, metodologías y contenidos vistos hasta el momento.

Requisitos obligatorios:

- php, html, css, mysql
- acceso a archivos
- formularios
- validación y control de usuarios y/o roles
- acceso a BBDD
- control de sesiones
- arrays unidimensionales, arrays multidimensionales
- documentación de uso

### Aspectos adicionales u optativos:

- subida al servidor
- cookies
- interacción con el sistema operativo
- envío de emails
- impresión de pedidos/documentos
- frontend y backend

Aplicación ejemplo:

***Aplicación web de pedidos a una cadena de restaurantes***

En las siguientes páginas se da una idea de una posible aplicación y las fases y tareas que requeriría.

Desarrollo web en entorno servidor

Ejemplo de aplicación completa en PHP

## Descripción general

- Aplicación web para una cadena de restaurantes.
- Los restaurantes de la cadena hacen pedidos al departamento de pedidos.
- Los usuarios de la aplicación son los restaurantes.
- Es como una tienda online.

## Funcionalidad

- Consultar las categorías.
- Consultar los productos.
- Añadir una o más unidades de un producto al carrito.
- Consultar el estado del carrito y eliminar productos de este.
- Realizar el pedido, introduciéndolo en la base de datos y enviando correos de confirmación al restaurante que hace el pedido y al departamento de pedidos de la empresa.

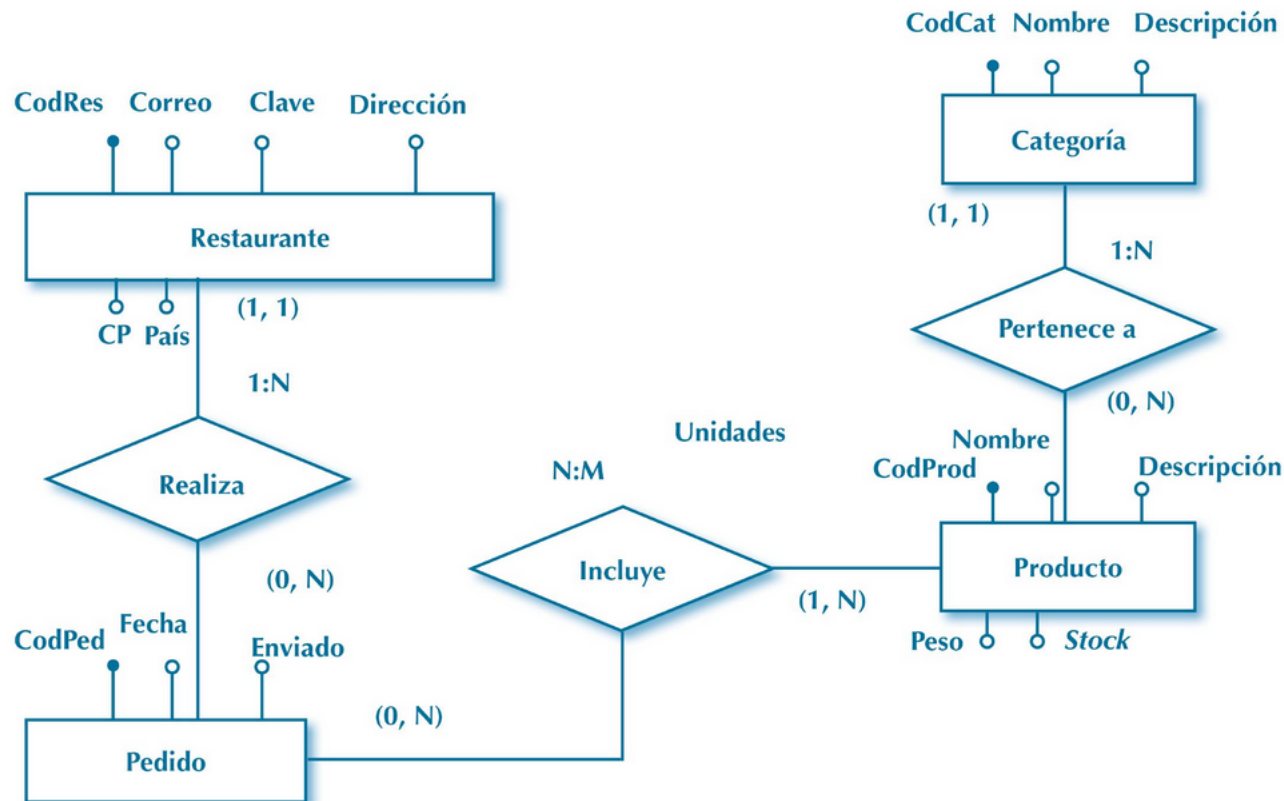
## Funcionalidad (2)

- Para acceder a la aplicación será necesario autenticarse.
- Se supone que en cada restaurante habrá un responsable de pedidos, que es quien tiene el usuario y la clave para acceder.
- No hay posibilidad de autorregistro. Los usuarios los crea el departamento de pedidos directamente en la base de datos.

## Consideraciones y limitaciones

- Sin pagos. Se trata de restaurantes de una misma empresa.
- Sin control de stock. Si algún producto se agota, en el envío se tramita igualmente.
- Sin panel de administración. Los usuarios, categorías y productos se introducen directamente en la base de datos.

## Esquema E-R



## Diseño de la base de datos

- Restaurantes(**CodRes**, Correo, Clave, Pais, CP, Ciudad, Dirección)
- Pedidos(**CodPed**, Fecha, Enviado, Peso, *Restaurante*)
- Productos(**CodProd**, Nombre, Descripción, Peso, Stock, *Categoria*)
- PedidosProductos(**CodPedProd**, *Pedido*, *Producto*, Unidades)
- Categorías(**CodCat**, Nombre, Descripción)

Las claves primarias aparecen en negrita, las ajenas en cursiva.



## Ejemplo de inserción de un pedido

### *Inserción de un pedido*

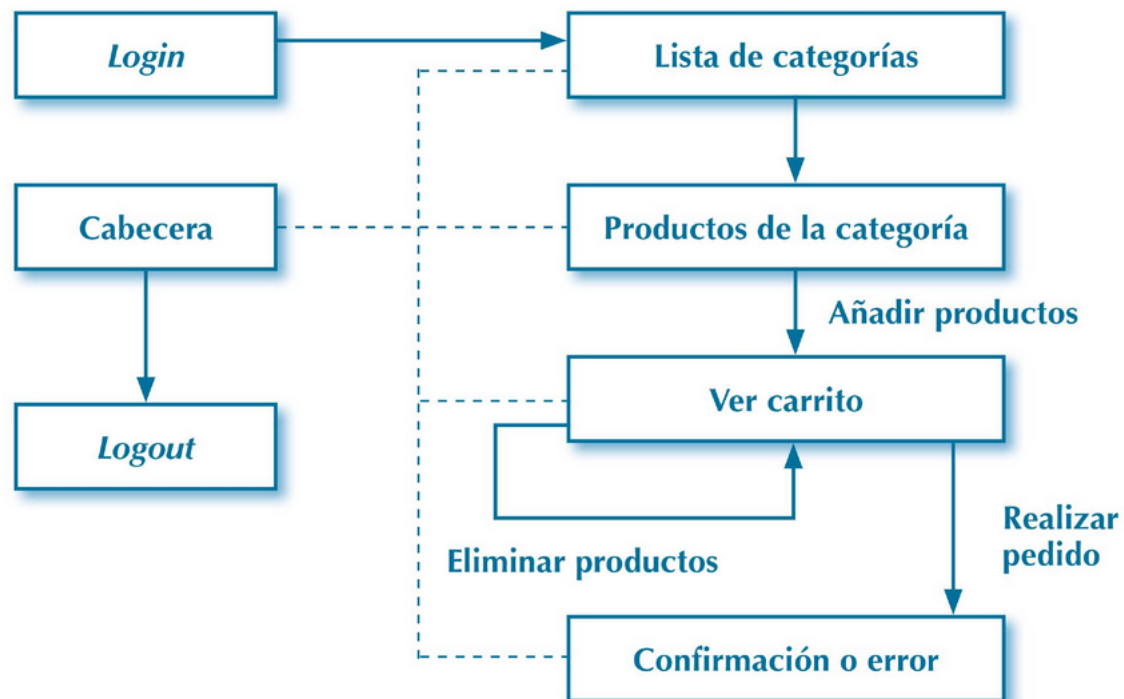
El usuario con código 1 hace un pedido de 100 unidades del producto con código 16 y 150 del producto con código 20. El pedido recibe el código 1.000. En la tabla pedidos se insertará:

CodPed	Fecha	Enviado	Ejemplo
1.000	01/09/2019	0	1

Como el pedido incluye dos productos diferentes, hay que insertar dos filas en la tabla PedidosProductos. Estas filas reciben los códigos 20.000 y 20.001.

CodPedProd	CodPed	CodProd	Unidades
20.000	1.000	16	100
20.001	1.000	20	150

## Mapa de pantallas



## Lista de categorías

- La página principal es una lista con las categorías presentes en las bases de datos.
- La lista muestra los nombres de las categorías con un vínculo.
- Si se sigue el vínculo se accede a la tabla de productos de esa categoría.

Usuario: madrid1@empresa.com [Home](#) [Ver carrito](#) [Cerrar sesión](#)

### Lista de categorías

- [Bebidas con](#)
- [Bebidas sin](#)
- [Comida](#)

## Tabla de productos de una categoría

- Muestra una tabla con los productos de la categoría.
- Una fila por producto.
- Al final de la fila incluye un formulario para comprar ese producto.
- El formulario contiene un campo para que el usuario introduzca el número de unidades y un campo oculto con el código.

**Bebidas con**

Bebidas con alcohol

Nombre	Descripción	Peso	Stock	Comprar	
Cerveza Alhambra tercio	24 botellas de 33cl	10	15	<input type="text" value="1"/>	<input type="button" value="Comprar"/>
Vino tinto Rioja 0.75	6 botellas de 0.75	5.5	10	<input type="text" value="1"/>	<input type="button" value="Comprar"/>

## El carrito de la compra

- Para el carrito de la compra se utiliza una variable de sesión, `$_SESSION[ 'carrito' ]`
- Es un array en el que habrá un elemento por cada producto (diferente) que se incluya en el pedido.
- La clave de los elementos es el código del producto, y su valor el número de unidades pedidas.
- Ejemplo: si se añaden 4 unidades del producto de código 100 y 30 del producto con código 200, el array tendrá dos elementos:  
`$_SESSION[ 'carrito' ][100]` con valor 4  
`$_SESSION[ 'carrito' ][200]` con valor 30

## Eliminar productos del carrito

- Cada fila de la tabla del carrito incluye un formulario para eliminar productos.
- El formulario contiene un campo para que el usuario introduzca el número de unidades que quiere pedir y un campo oculto con el código del producto.

Usuario: madrid1@empresa.com [Home](#) [Ver carrito](#) [Cerrar sesión](#)

---

**Carrito de la compra**

Nombre	Descripción	Peso	Unidades	Eliminar
Sal	20 paquetes de 1kg cada uno	20	1	<input type="text" value="1"/> <input type="button" value="Eliminar"/>

---

[Realizar pedido](#)

## Añadir productos

- Pseudocódigo para añadir *unidades* unidades del producto con código *cod* al carrito:
  - Comprobar si el array ya contiene un elemento con clave *cod*. Es decir, si existe `$_SESSION[ 'carrito' ][ cod ]`.
  - Si no lo tiene, se crea con valor *unidades*.
  - Si ya lo tiene, se actualiza su valor sumándole *unidades*.

## Eliminar productos

- Pseudocódigo para eliminar *unidades* unidades del producto con código *cod* al carrito:
  - Comprobar si el array ya contiene un elemento con clave *cod*. Es decir, si existe `$_SESSION[ 'carrito' ][ cod ]`.
  - Si ya existe un elemento con esa clave, se actualiza su valor restándole *unidades*.
    - Si tras la resta el valor es menor o igual que 0, se elimina el elemento del array.
  - Si no existe el elemento en el array, no hay que hacer nada.



## Realización de un pedido

Al realizar un pedido:

- Se insertan las filas correspondientes en las tablas de Pedidos y PedidosProductos.
- Se envía un correo con los datos del pedido tanto al restaurante que lo realiza como al departamento de pedidos.
- Se vacía el carrito.

## Cabecera

Una vez se inicia sesión, hay una cabecera común en todas las páginas.

Usuario: madrid1@empresa.com [Home](#) [Ver carrito](#) [Cerrar sesión](#)

---

## Envío de correos

- Al realizar el pedido se envía un correo de confirmación.
- Se utiliza una cuenta de Gmail para enviar correos.

**Sistema de pedidos**

to madrid1@empresa.com, pedidos@empresafalsa.com ▼

### Pedido nº 74

**Restaurante:** [madrid1@empresa.com](mailto:madrid1@empresa.com)

Detalle del pedido:

Nombre	Descripción	Peso	Unidades
Agua 0.5	100 botellas de 0.5 litros cada una	51	1
Vino tinto Rioja 0.75	6 botellas de 0.75	5.5	1

## Descripción de los ficheros

Ruta	Descripción	Parámetros	Redirige a
login.php	Formulario de <i>login</i>	\$_GET['redirigido'] \$_POST['usuario'] \$_POST['clave']	login.php?error=TRUE categorias.php
logout.php	Cierra la sesión		
sesiones.php	Comprueba que el usuario haya iniciado sesión correctamente		login.php (si no se ha iniciado sesión)
categorias.php	Muestra la lista de categorías con vínculos a productos.php?categoria=codigo		
cabecera.php	Cabecera con vínculos para ver el carrito, las categorías o cerrar sesión		
productos.php	Muestra los productos de la categoría, permite añadir al carro de la compra	\$_GET['categoria'], el código de la categoría	
carrito.php	Muestra el carro de la compra, permite quitar productos y confirmar el pedido		
anadir.php	Añade productos al carrito	\$_POST['cod'] \$_POST['unidades']	carrito.php
eliminar.php	Elimina productos del carrito	\$_POST['cod'] \$_POST['unidades']	carrito.php
procesar_pedido.php	Inserta el pedido en la base de datos, envía correos de confirmación y muestra mensajes de error o éxito		
bd.php	Para agrupar las funciones de la base de datos		
correo.php	Funciones para enviar correo		

## Control de acceso

- Al abrir sesión se crea una variable de sesión con el correo y el código del usuario.
- Para controlar el acceso, se comprueba la existencia de la variable.

## 5.- BIBLIOGRAFÍA

### Desarrollo Web En entorno Servidor

Editorial Síntesis

Xabier Ganzábal García

isbn: 978-54-917183-6-9

#### Resumen

- La aplicación permite que los restaurantes de una cadena realicen pedidos al Departamento de Pedidos.
- Muestra los productos disponibles organizados por categorías.
- El carrito de la compra es una variable de sesión. Es un *array* en que la clave de cada elemento representa el código del producto, y su valor, las unidades pedidas.
- El carrito se manipula a través de formularios que envían el código del producto y las unidades que hay que añadir o eliminar.
- Al añadir productos al carrito, si no hay ya un elemento con ese código, se crea. Si lo hay, se suman las unidades.
- Al eliminar productos del carrito hay que comprobar si quedan 0 o menos unidades para eliminar el elemento correspondiente.
- Al pulsar el vínculo de realizar pedido, los datos del carrito se utilizan para modificar la base de datos.
- El pedido se realiza como una transacción; si hay algún error con la base de datos, se deshacen las operaciones anteriores.
- Si el pedido se realiza con éxito, se envían correos de confirmación al Departamento de Pedidos y al restaurante que lo realiza.
- Hay muchas ampliaciones sencillas por hacer: control de *stock*, panel de administración, contraseñas encriptadas...