

Práctica Final

2 DAW



Restaurante TPV

Pedro Sandoval Martínez

Índice

[Primer Sprint 2](#_Toc182487113)

[Diseño y creación de la base de datos 2](#_Toc182487114)

[Esquema de Archivos 3](#_Toc182487115)

[Inicio de sesión y ruta de acceso 3](#_Toc182487116)

[Menús de opciones 4](#_Toc182487117)

[Spring 2 5](#_Toc182487118)

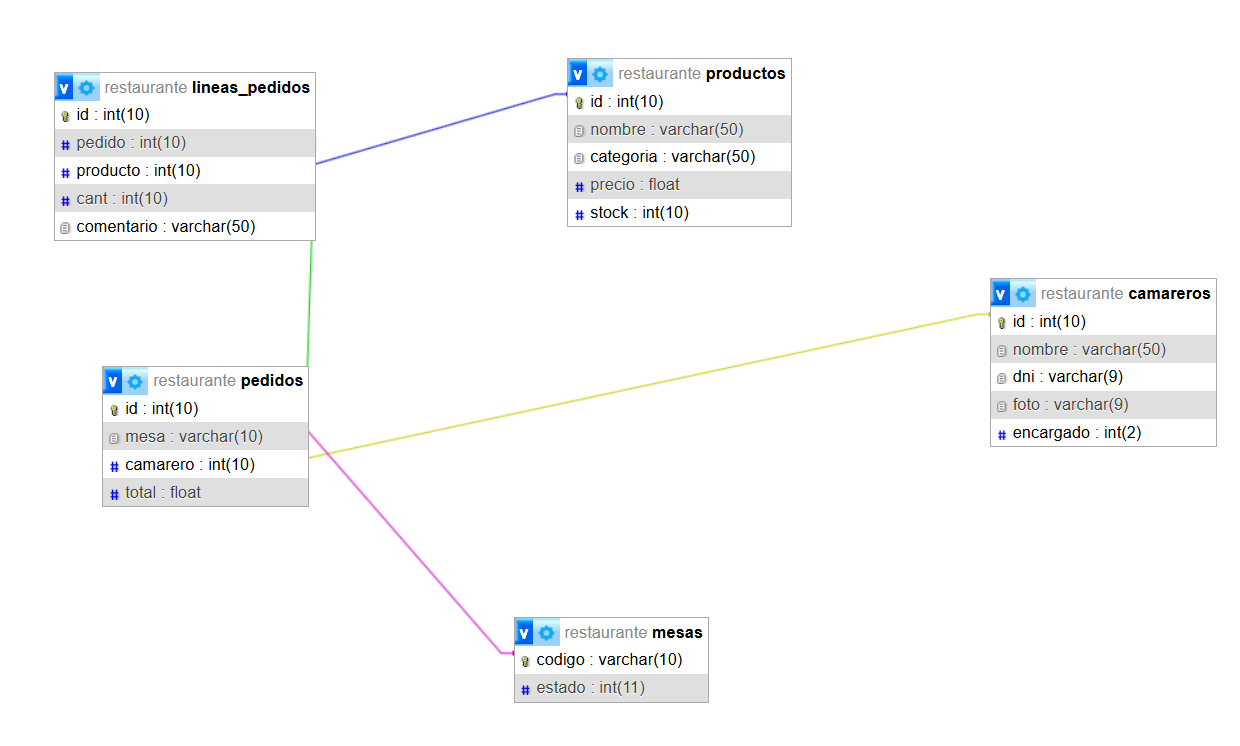
[Gestión y visualización de mesas 5](#_Toc182487119)

[Creación de pedidos 6](#_Toc182487120)

# Primer Sprint

## Diseño y creación de la base de datos

Primer diseño de la base de datos. Centrada sobre todo en la tabla camareros, en la cual he comenzado a trabajar. En principio el id del camarero estará en el pedido para conocer quién registra el mismo.



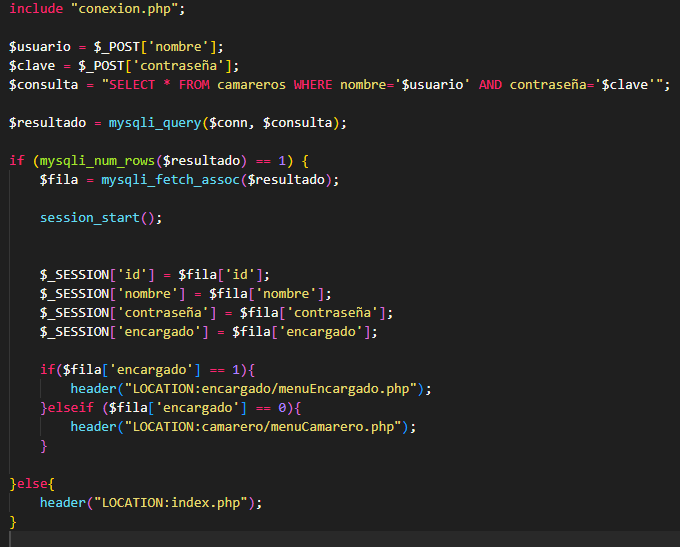
## Esquema de Archivos

De momento he separado la visión de camarero y encargado.

Los archivos principales que conectan con la base de datos se encuentran en la carpeta raíz, al igual que los estilos del index (inicio de sesión) y el archivo que utilizo para validar la sesión.

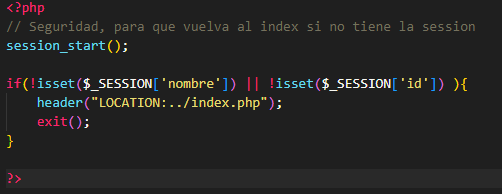
## Inicio de sesión y ruta de acceso

Previamente insertado el primer camarero (admin), comprobamos que el usuario que intenta acceder se encuentre en la base de datos.



Si el usuario, es encargado (atributo encargado = 1), la página es redirigida al apartado de encargado. En cambio, si el atributo es 0, te lleva a la vista del camarero.

La seguridad del sitio web es comprobada por este archivo, que comprueba en cada una de las páginas la variable de sesión:



## Menús de opciones

En la vista de **camarero**, únicamente encontramos la opción de entrar al salón donde visualizamos las mesas, y dentro de ellas la toma de los pedidos.

El menú de **encargado** lo presentamos inicialmente así. Entre sus opciones encontramos la modalidad de añadir camareros, el listado de los mismos (sus datos), añadir nuevos productos y visualizarlos.



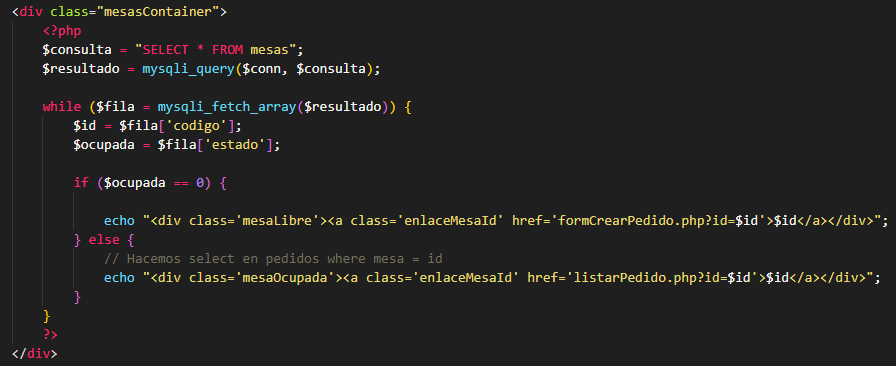
Además, introduzco un enlace para acceder a la vista de camareros por si el encargado lo necesitase.

# Spring 2

## Gestión y visualización de mesas

Breve explicación: las mesas las utilizamos como contenedor del pedido, para localizarlo en el salón y poder acceder a las opciones (crear pedido y listarlo).

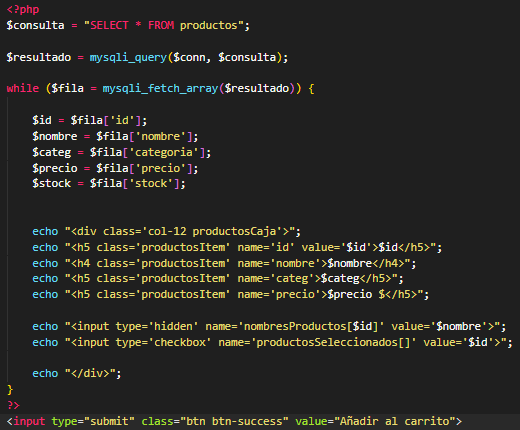
Archivo salón.php :



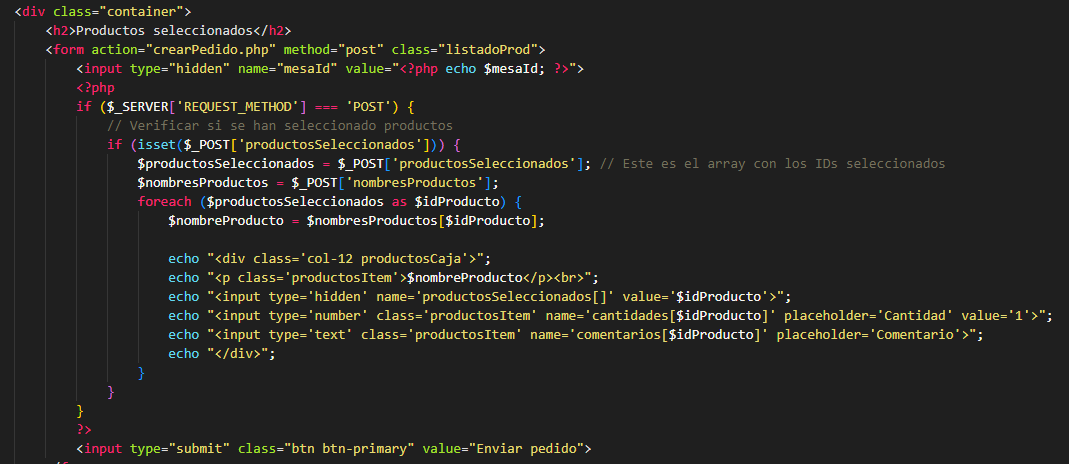
Mostramos las mesas que se encuentren en la base de datos, las que tengan estado 1 (ocupada) se mostrarán con una clase distinta, para que cambie el diseño y sea más fácil localizarlas en el salón.

## Creación de pedidos

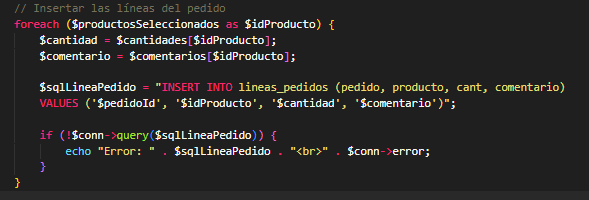
Una vez que entramos a la mesa, mostramos en un formulario la consulta de los productos. Los artículos que seleccionemos pasarán a otro formulario (carrito).



En el mismo archivo (formCrearPedido), hay otro formulario que usamos a modo de carrito, donde nos llevamos el array de productos seleccionados (con el checkbox).



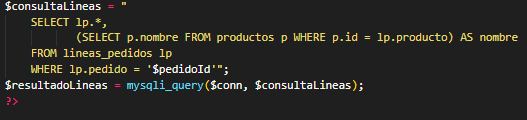
Este formulario es el que lleva a crearPedido, donde finalmente se hace la inserción en líneas\_pedidos. Para ello he tenido que llevarme el id del pedido, de la mesa y de los artículos seleccionados. Todo ello en forma de array con cada uno de los elementos ordenados por su id.



Tras crear el pedido y las líneas del pedido, el estado de la mesa pasa a ser ocupado.

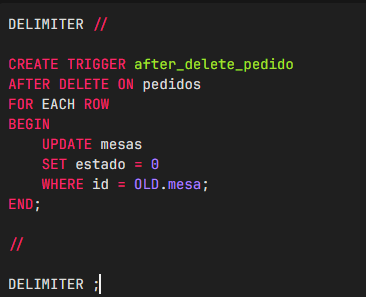
## Abrir mesas activas

Una vez que la mesa aparece ocupada, si accedemos a ella desde el salón nos llevará a la página donde se listan las líneas del pedido que hay alojado en el mismo. Para ello he creado una consulta con producto cartesiano



Trigger

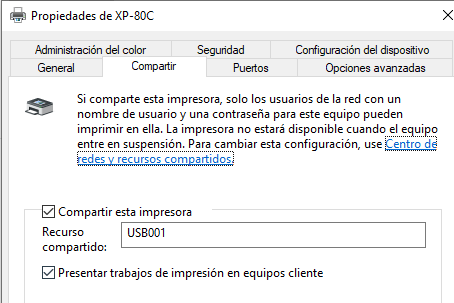
Cuando se elimine un pedido



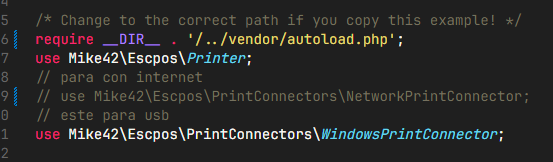
# Conexión con la impresora

Inicialmente hice las pruebas conectando la impresora (modelo XP-80C) por usb y posteriormente instalando los drivers pertinentes, en modo local con los propios archivos que nos facilitó el profesor D. Fernando Ureña.

El nombre que le otorgues es irrelevante, sin embargo, un paso indispensable es la configuración del puerto al que está conectada. En Windows (preferencias de impresora) tuve que activar el uso compartido de la misma, asignándole el puerto en el que será reconocida por nuestro servidor y por tanto en nuestros archivos PHP.



Por supuesto otro requisito indispensable es asegurarnos que el composer y el vendor previamente instalados en el repositorio están enlazados con el archivo donde creamos el ticket, que no haya errores en la ruta.



Esto incluye en nuestro archivo las librerías que necesitamos.