



MEMORIA DE LA PRÁCTICA DE PHP

Grid & Growler



16 DE NOVIEMBRE DE 2025

2º DAW
Eduardo Piquer Sánchez



Índice:

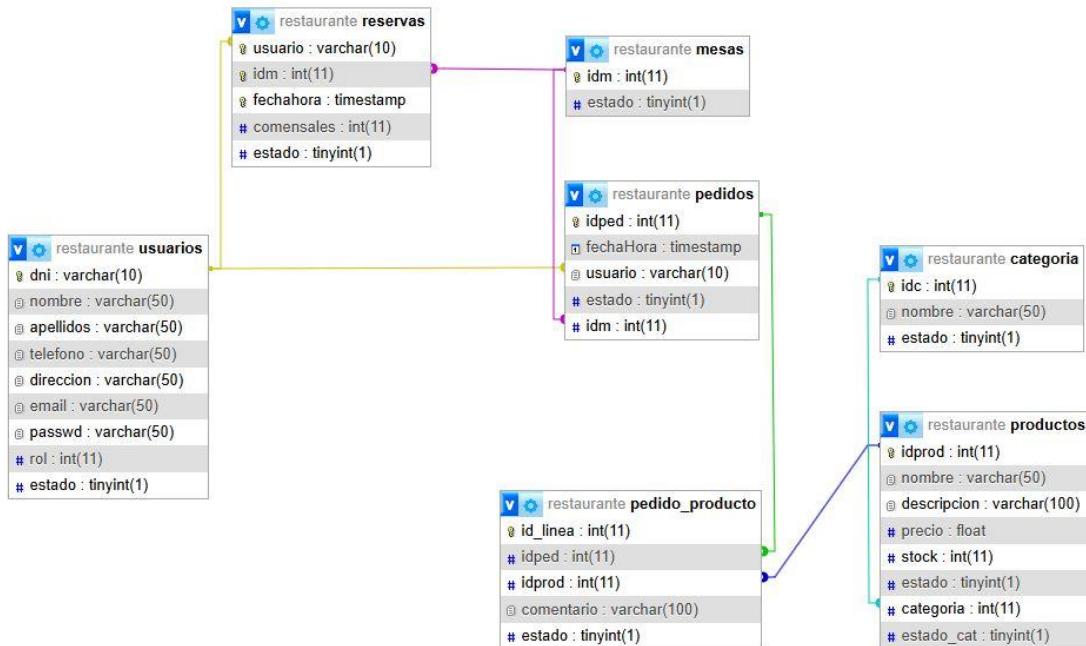
1.	Base de datos relacional:	2
2.	Sprint 1:	3
3.	Sprint 2:	5
4.	Problemas encontrados:	8



1. Base de datos relacional:

Para crear la página web del restaurante **Grid & Growler** he creado y relacionado las siguientes tablas:

- Usuarios (dni, nombre, apellidos, telefono, direccion, email, passwd, rol, estado)
- Reservas (usuario, idm, fechaHora, comensales, estado)
- Productos (idprod, nombre, descripción, precio, stock, estado, categoría, estado_cat)
- Mesas (idm, estado)
- Pedidos (idped, fechaHora, usuario, estado, idm)
- Categoría (idc, nombre, estado)
- Pedido_producto (id_linea, idped, idprod, comentario, estadog)





2. Sprint 1:

Durante este primer sprint, comencé generando una plantilla base en **HTML5**, pensada para adaptarse a las necesidades específicas de la práctica. A partir de esa estructura inicial, fui realizando distintas modificaciones y ajustes de diseño para adaptarla a mi propio estilo y al enfoque que quería darle al proyecto. Además, me apoyé en **Bootstrap** para asegurar que la plantilla fuera totalmente responsive, garantizando así una correcta visualización en diferentes dispositivos y resoluciones de pantalla.

Después, he creado la base de datos, definiendo todas las tablas, sus relaciones y atributos necesarios para el correcto funcionamiento del sistema. Una vez tuve la base de datos preparada, he implementado la pantalla de inicio. En ella, según el rol del usuario, la página redirige automáticamente a la sección correspondiente.

```
Bootstrap > login.php > ...
1  <?php
2  session_start();
3  include("includes/conexion.php");
4
5  if ($_SERVER['REQUEST_METHOD'] === 'POST') {
6      if (isset($_POST['dni']) && isset($_POST['passwd'])) {
7
8          // Creo variable con dni del usuario logueado
9          $dni = $_POST['dni'];
10         $pass = $_POST['passwd'];
11
12         // Creo consulta con los datos del usuario logueado
13         $consulta = "SELECT * FROM usuarios WHERE dni='$dni' AND passwd='$pass'";
14         $result = mysqli_query($conn, $consulta);
15         echo mysqli_error($conn);
16         mysqli_close($conn);
17
18         if (mysqli_num_rows($result) == 1) {
19             //Usuario registrado
20             $row = mysqli_fetch_assoc($result);
21
22             $_SESSION['rol'] = $row['rol'];
23
24             // Redirigimos según el rol
25             switch ($_SESSION['rol']) {
26                 case 1:
27                     header('Location: encargado/gestion_personal.php');
28                     exit();
29                 case 2:
30                     header('Location: camarero/mesas.php');
31                     exit();
32                 case 3:
33                     header('Location: cliente/elegir_mesa.php');
34                     exit();
35                 default:
36                     // Si hay un rol raro, lo mandamos al login
37                     header('Location: index.php');
38                     exit();
39             }
40         } else if (mysqli_num_rows($result) > 1) {
41             //Fallo de integridad en la bbdd;
42         } else {
43             $_SESSION['error'] = "Usuario no registrado o datos incorrectos";
44             header('Location: index.php');
45             exit();
46         }
47     }
48 }
```



También diseñé los menús de navegación para cada tipo de usuario, aunque en este sprint todavía no tienen funcionalidad completa.

A continuación, incorporé las medidas de seguridad básicas y el sistema de cierre de sesión, para garantizar un acceso controlado. Luego, desarrollé el sistema de registro de clientes, permitiendo que nuevos usuarios puedan crear su cuenta en la plataforma. La siguiente imagen muestra la seguridad del cliente.

```
Bootstrap > cliente > seguridad_cliente.php
1  <?php
2  // Comprobamos que el usuario ha iniciado sesión y su rol es '3' (camarero)
3  if (!isset($_SESSION['rol']) || $_SESSION['rol'] != '3') {
4      header('Location: ../index.php');
5      exit();
6  }
7 ?>
```

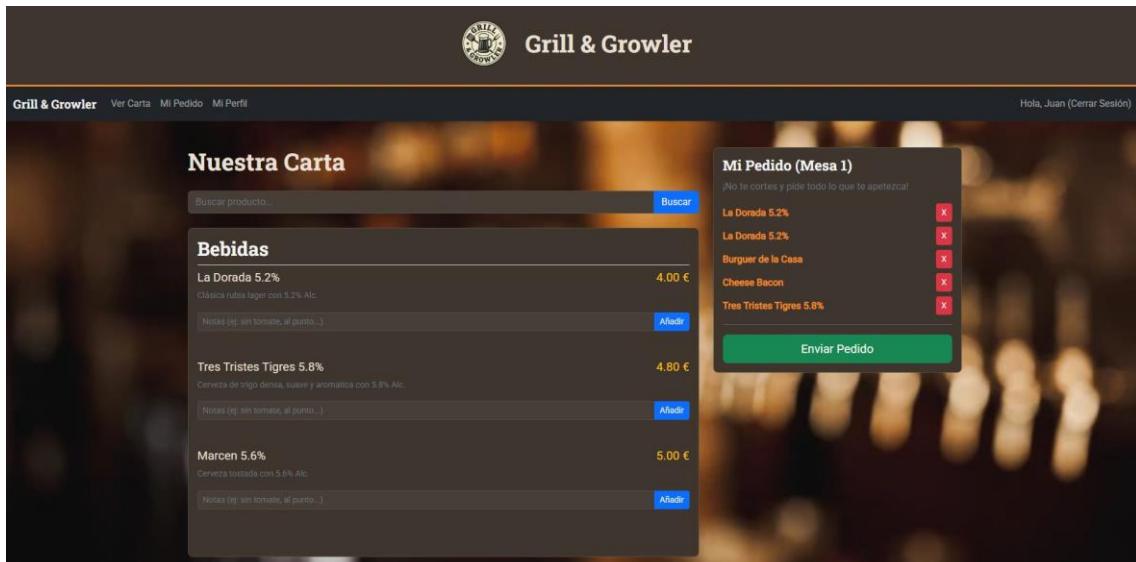
En la parte del encargado, he creado las páginas para crear, eliminar y modificar categorías, así como para añadir o actualizar productos y su stock.

The screenshot shows the Grill & Growler website with a dark header. The header includes the logo, the restaurant name 'Grill & Growler', and a navigation bar with links for 'Mesas Activas', 'Informes', and 'Gestión'. A dropdown menu titled 'Gestión de Personal' is open, showing options like 'Personal', 'Cílios', 'Productos y Stock', and 'Categorías'. Below this, a table titled 'Personal A' lists one employee: 'Juan Pérez' with DNI '12345678A', phone '675675675', address 'C/ perico', email 'pepe@gmail.com', role 'Camarero', and a red 'Suspender' button. A green 'Añadir Nuevo' button is visible at the top right of the table area.



3. Sprint 2:

Durante este segundo sprint, diseñé y desarrollé la interfaz para que los clientes pudieran seleccionar la mesa y el número de comensales, así como elaborar su pedido, asegurándome de que sólo se mostraran los productos con stock disponible. Me encargué de implementar un sistema de control del stock en tiempo real, de modo que cada pedido confirmado descontara automáticamente la cantidad pedida y evitará confirmar pedidos si el stock era insuficiente, mostrando un mensaje pertinente.





Aseguré que los clientes pudieran añadir productos a un pedido ya iniciado, sin modificar aquello que ya habían enviado, permitiendo así múltiples rondas, y añadí la reciente funcionalidad para que cada producto pueda incluir notas específicas. También integré un buscador para facilitar que los clientes filtren productos por el nombre del producto.

Producto	Notas	Estado	Precio
La Dorada 5.2%		Pendiente	4.00 €
La Dorada 5.2%		Pendiente	4.00 €
Burguer de la Casa		Pendiente	11.50 €
Cheese Bacon		Pendiente	12.50 €
Tres Tristes Tigres 5.8%		Pendiente	4.80 €
Total a Pagar: 36.80 €			

Producto	Notas	Estado	Precio
La Dorada 5.2%		Pendiente	4.00 €
La Dorada 5.2%		Pendiente	4.00 €
Burguer de la Casa		Pendiente	11.50 €
Cheese Bacon		Pendiente	12.50 €
Tres Tristes Tigres 5.8%		Pendiente	4.80 €
Burguer de la Casa		Pendiente	11.50 €
Marcen 5.6%		Pendiente	5.00 €
La Dorada 5.2%		Pendiente	4.00 €
Total a Pagar: 57.30 €			



Por parte del camarero, desarrollé la funcionalidad para visualizar todas las mesas activas y actualizar el estado de cada pedido, pudiendo cambiarlo de “pedido” a “servido”. Implementé también un bloqueo para que un cliente no pueda seleccionar una nueva mesa mientras la actual no esté pagada y liberada, asegurando el correcto control del sistema.

Grill & Growler (Camarero) Ver Mesas Activas

Hola, Benito (Cerrar Sesión)

Estado de las Mesas

Mesa 1 Ocupada Comensales: 5	Mesa 2 Libre	Mesa 3 Libre
------------------------------------	-----------------	-----------------

Grill & Growler (Camarero) Ver Mesas Activas

Hola, Benito (Cerrar Sesión)

Gestionando Mesa 1

Productos Pedidos	Notas	Estado	Acción
La Dorada 5.2%		Servido	Servido
La Dorada 5.2%		Servido	Servido
Burguer de la Casa		Servido	Servido
Cheese Bacon	Pendiente		Marcar Servido
Tres Tristes Tigres 5.8%	Pendiente		Marcar Servido
Burguer de la Casa	Pendiente		Marcar Servido
Marcon 5.6%	Pendiente		Marcar Servido
La Dorada 5.2%	Pendiente		Marcar Servido

Generar Cuenta
El cliente ha pedido la cuenta. Pulsa aquí para imprimir el ticket.

Generar Cuenta y Ticket



4. Problemas encontrados:

El primer contratiempo que he tenido ha sido al crear y mostrar la lista a la que se van agregando los productos antes de ser enviados a cocina y barra. Al final he creado un array de sesión, al que se va añadiendo uno a uno cada producto, y posteriormente se muestra en el lateral derecho de la página.

También he tenido algún problema para generar la búsqueda de productos del cliente, ya que tengo el menú separado por categorías, y cuando buscaba un producto me seguían apareciendo todas las categorías. Al final generé el código de búsqueda fuera del código que generaba las categorías, separados por un if/else.