



Saludos a todos los que me lean;

Mi proyecto se basa en un restaurante, denominado Le Macià, por el instituto.  
Comentaré todas las pautas que hay y como funcionan.

Para empezar podemos abrir [index.php](#) para comenzar y ver la portada, del cual consta de:

- Header
  - Menú principal que equivale a la franja negra del top que cambia por la hamburguesa al cambiar la resolución/dispositivo.
    - Se muestra el icono de Perfil/Carrito/Fecha y los apartados Inicio, Ayuda y Contacto.
    - Banners, digo en plural porque hay 2 y dependiendo de la resolución/dispositivo uno se muestra y el otro se oculta, ya que en versiones mas pequeñas no se veía bien el primer banner que hay.
    - Menú: Aquí es una simple lista `<ul><li>` con efectos por JavaScript y CSS para que los hovers se vean de modo lento. Su función es simplemente redirigir a las páginas que dice. (Entrantes, Comidas, Bebidas, Postres..).
- Main
  - Sección 1
    - Consta de varias imágenes, de las cuales tres de ellas (El chocolate, la hamburguesa y la tabla con la hamburguesa) he jugado con los valores relative y absolute para dejarlos de esa forma, mientras que las otras tres del centro están con un flex, además de tener un bonito efecto de que se le da la vuelta y muestra un texto al poner el cursor encima o hacerle click.
  - Sección 2
    - Consta de 2 gifs a los lados hecho por mi con canva.com y otros 3 en el centro pero que funcionan mediante una función de JavaScript para hacer un slider, por lo que van alternando tres gifs todo el tiempo nada mas se cargue la página con `window.onload`.

- Sección 3

- Consta de un grid de 3x3. En principio las imagenes iban a ser clickables y se aumentaba, pero tras muchas pruebas que no funcionaba bien, decidí dejarlo como un tablón de fotos y ya.

Seguimos con la página de **login.php**, como observarán pueden ver que se introduce un correo, una contraseña en los inputs y dos botones, Enviar y Regístrate. Empezando por lo más facil, el botón Regístrate redirecciona a la página **registro.php**, que hablaremos después.

Los campos están pasados por funciones de comprobación por si se escribe mal el usuario o la contraseña o está vacío. Estas comprobaciones se hacen mediante JavaScript. Además de que si no existe en la base de datos, en la tabla registro, es imposible entrar.

Una vez iniciemos sesión nos redirige a la página de **inicio-sesion.php** (esta página va anclada a inicio-sesion-backend.php) que es donde nos mostrará tres botones.

- Si entramos con la cuenta [admin@gmail.es](mailto:admin@gmail.es) - admin123 tendremos los botones Mis datos (que simplemente muestra los datos de tu cuenta, más no la contraseña porque no lo veía bien), Administrar, que este botón solo se verá en caso de entrar con esta cuenta y cerrar sesión. Al haber iniciado sesión se crea una sesión (PHP) con el correo y nombre. El correo es lo que utilizaremos de identificador para hacer todo ya que es unique. Esto se aplica tanto en la cuenta de admin como usuario. No se puede acceder a inicio-sesion.php si no esta logeado por la sesión que se ha creado y lo redirige de vuelta a login.php. Toda página que necesite saberse el correo como identificador, tiene que iniciarse una sesión mediante el login.php, de lo contrario todas las páginas redigirán a login.php (carrito.php, pedidos.php).
- Si entramos como usuario normal, que la podeis adquirir registrandoos o usando la cuenta de [user@gmail.es](mailto:user@gmail.es) - user123, no podreis ver el botón de admin, en lugar de ese botón vereis el de carrito, además de mis datos y cerrar sesión. Al cerrar sesión se esta destruyendo la sesión y en la tabla login cambia de conectado a 0. Así se puede saber quien esta logeado y quien no. 1 es conectado y 0 desconectado.

En la página de registro.php (Botón Regístrate de login.php) podremos ver que hay campo nombre, correo electrónico y contraseña, la cuál tiene unas validaciones mediante expresiones regulares en el index.js para comprobar si se meten correctamente los datos. En caso de que no esté correcto mostrará un mensaje y evita la subida del formulario. Esto lo hice más que nada porque al darle al botón submit, la página se refresca y desaparecen los mensajes.

A continuación si clickamos en el carrito (**carrito.php**) odremos ver la lista de productos que hemos añadido al carrito de los apartados comida, bebida, entrantes, postres. Al hacer click podemos observar un mensaje con “No hay nada en el carrito” o los productos que hemos seleccionado previamente con

una tabla el cual contiene una X al lado de cada producto por si nos arrepentimos y no queremos añadirlo. Además sale una lista de todos los productos, la cantidad, precio y el total. Se puede ver el resumen del carrito haciendo click en ticket (Ojo! La librería FPDF se ha quedado obsoleta y solo funciona con Edge Microsoft, almenos en mi caso, si quereis probarlo con otro navegador adelante, pero es posible que lo veais en blanco)

Al darle a Realizar pedido lo que estamos haciendo es enviar ese pedido de todos los productos al admin, que con el botón de “Administrar” que se ve al logearse como admin.

Una vez clickado en Realizar pedido, se borra de la base de datos de “carrito” y se añade a “pedidos”, visibles unicamente para el admin.

Tanto el usuario como el admin puede borrar mediante la X cualquier producto de la lista. Este archivo de carrito.php, va anclado al archivo **eliminar\_pedido.php** e **insertar\_pedidos.php**, que como dice el nombre son la parte “backend” para insertar o eliminar pedidos.

Para añadir productos al carrito, hay que entrar en las páginas **frontend-bebidas.php**, **frontend-comidas.php**, **frontend-entrantes.php** o **frontend-postres.php**. Estas páginas van ancladas **backend-comidas** y **backend-pedidos**. Estos archivos están divididos en tipos, es decir, entrantes 0, comidas 1, bebidas 2, postres 3 y dependiendo de qué archivo se abra, hace una consulta para que muestre solo los datos del tipo que es, por ejemplo si abrimos frontend-bebidas.php hay una consulta que dice `$comida = "SELECT * FROM comida where tipo = 2";` tras eso, mostrará los datos que debe enviando esa consulta a **backend-comidas.php**.

Estas páginas consta de un grid de productos que se muestra el nombre, la imagen, descripción y precio desde la base de datos y se puede eliminar o insertar más productos sin que afecte visualmente por el grid que hay.

Todo usuario puede ver el producto, entra, clicka sobre el producto deseado y puede ver el precio, porque no necesariamente se necesita registrarse e iniciar sesión en una página para ver un producto, pues aquí igual, puede ver el producto más no añadirlo al carrito, ya que no hay una sesión. Por lo tanto para poder ver el botón para comprar hay un display none si no hay sesión iniciada.

La ventana que aparece en medio al clickar sobre un producto está con una animación de fade-out de JavaScript y CSS.

Al clickar en Añadir mostrará una ventana emergente diciendo añadido al carrito. Esta ventana se muestra gracias a la librería de Swall de CSS, que es simplemente para este tipo de ventanas, ya que los alerts no se pueden modificar.

Y con esto concluyo la documentación de mi proyecto. Lamento que visualmente no esté mejor, lo he repetido varias veces intentando ponerla bonita, pero no tengo imaginación, sin embargo es totalmente funcional y con esto me he dado cuenta que me gustaría dedicarme al backend.

Gracias por su atención, atentamente Vanessa Rubio Sánchez

Librerías usadas:

- Bootstrap
- Swal
- FPDF
- JQuery

PD: Intenté usar la librería PHPMailer, pero no me funcionaba por los puertos.