



**JUAN DAVID SERVELLON**

**ALEJANDRO SAINZ**

**LAILA FERNÁNDEZ**

***1DAW***

***Trabajo grupal – Burger Factory***

## ÍNDICE

NUESTRA IDEA .....	3
INDEX.....	4
Botones .....	4
Header .....	4
Section .....	5
Div .....	6
Otros .....	7
STYLES.....	8
Sin @media.....	8
Con @media .....	9
SCRIPT .....	10
Funciones.....	10

## NUESTRA IDEA

Hemos creado una pagina web sobre una tienda de hamburguesas con la idea de poder hacer una reserva de las hamburguesas, añadiendo la hora de llegada y dejarlo todo pagado, de esta forma cuando usted llegue con sus amigos, familiares, etc... No tendrá tiempo de espera para disfrutar de nuestra mejor comida, además también tenemos la opción de pedirlo y reservar y cuando usted llegue a la hora que ha reservado, poder llevártelo sin la necesidad de comértelo ahí, podrás disfrutar en cualquier zona de nuestras maravillosas comidas.

Contamos con un amplio menú, 4 hamburguesas de carne, 2 hamburguesas para vegetarianos, 3 hamburguesas de pollo y además 3 menús de bocadillo, bebida y fruta para los universitarios ya que nuestro local se encuentra enfrente de las universidades y sabemos que a más de uno le va a venir bien.

Tenemos la opción de contactar con nosotros a través de correo electrónico o a través de una llamada, con un horario específico, que dentro de el resolveremos todas vuestras dudas, reservas, peticiones...

Si es de día y no ve bien la pantalla por el sol, podrá ponerlo en modo oscuro para que le ayude a la visibilidad, y viceversa, si es de noche y no lo ve bien lo podrá poner en modo claro.

# INDEX

## Botones

Hemos creado diferentes tipos de botones:

Botón para cambiar el estilo de la página, que pase de blanco a negro y viceversa.

```
<button id="cambiar_estilos"> </button>
```

Botón para el carrito de la compra, al darle en el botón te saldrá la información de lo que vas a comprar.

```
<button class="botonCompra" id="carro_compra">Carrito</button>
```

Botón de agregar la comida, cuando le des al botón agregar, se te añadirá al carrito el producto seleccionado.

```
<button class="botones_agregar" id="h1"> AGREGAR </button>
```

Botón de contacto 1 - Botón con nuestro número de teléfono para llamar directamente.

```
<button> TLF: +34 640 37 52 14 </button>
```

Botón de contacto 2 – Botón con nuestro correo para escribirnos lo que necesites directamente.

```
<button> CORREO: burgerfactory2025@gmail.com </button>
```

Botón, al rellenar el formulario para la reserva si pinchas en este botón automáticamente se te confirmara la reserva.

```
<button type="submit">Reservar</button>
```

## Header

En la parte superior de la página esta nuestro “header” que básicamente hemos añadido una imagen de nuestro logo y un “div” para poner imágenes de las redes sociales desde las que nos puede seguir la gente.

También tenemos un navegador en la parte derecha en la que está el botón que hemos visto anteriormente, el de cambio de estilo y las referencias a los diversos apartados de nuestra página.

```

<header>
  <div class="logo">
    
    <div class="RRSS">   </div>
  </div>
  <nav>
    <button id="cambiar_estilos"> </button>
    <a href="#" class="nav-link"> Inicio </a>
    <a href="#" class="nav-link"> Sobre nosotros </a>
    <a href="#" class="nav-link"> Contacto </a>
  </nav>
</header>

```

## Section

Tenemos tres “Section” uno para los párrafos donde hablamos de quienes somos y que el “nav” nos lleve directamente aquí.

```

</section>
<section id="somos">
  <div class="quien-somos">
    <h2 class="parrafos_modificar">Somos BurgerFactory</h2>
    <p class="parrafos_modificar">Somos un nuevo sitio de hamburguesas que lleva en proceso MÁS DE DOS AÑOS, por fin podemos presentarnos como es debido, las mejores hamburguesas que vas a probar están aquí, una variedad muy extensa, tanto para veganos como de carne y pollo, todo lo que busques lo vas a tener aquí. Además con la primera reserva te puedes llevar un 15% en todo lo que pidas ese día.
    </p>
  </div>
</section>

```

Otro para el contacto con nosotros, otra vez para que el “nav” nos lleve directamente aquí, en donde tenemos toda la información de como poder contactar con nosotros.

```

<section id="contacto">
  <h1 class="parrafos_modificar"> Contacta con nosotros </h1>
  <div class="contacto1">
    <h3> Numero de telefono </h3>
    <p class="parrafos_modificar"> Contacta con nosotros a traves de nuestro numero de telefono con el que estamos operativos desde las 10:00 hasta las 20:00, atenderemos desde cualquier tipo de duda hasta una reserva.
    </p>
    <button> TLF: +34 640 37 52 14 </button>
  </div>
  <div class="contacto2">
    <h3> Correo electronico </h3>
    <p class="parrafos_modificar"> En caso de que no puedas contactar con nosotros a traves de nuestro numero de telefono, tenemos la posibilidad de contactara traves de un correo electronico, en este no se podra hacer reservas pero podras resolver cualquier duda.
    </p>
    <button> CORREO: burgerfactory2025@gmail.com </button>
  </div>
  <div class="contacto3"></div>
</section>

```

Y el último que es para la reserva, en el tenemos un formulario con la opción de agregar el nombre, la fecha, la hora, el número de teléfono y algún comentario, también está el botón visto anteriormente para realizar la reserva.

```

<section id="reserva">
  <div class="container">
    <h2>Reserva tu Mesa</h2>
    <form>
      <label for="name">Nombre:</label>
      <input type="text" id="name" name="name" required>

      <label for="people">Número de Personas:</label>
      <input type="number" id="people" name="people" min="1" required>
    </form>
  </div>
</section>

```

```

<label for="date">Fecha:</label>
<input type="date" id="date" name="date" required>

<label for="time">Hora:</label>
<input type="time" id="time" name="time" required>

<label for="phone">Teléfono:</label>
<input type="tel" id="phone" name="phone" required>

```

```

<button type="submit">Reservar</button>
</form>
</div>
</section>

```

(La imagen esta dividida en 3, cada imagen es la continuidad de lo anterior)

## Div

Tenemos 12 “div” como este, creados para todos los menús, en ellos tenemos un “img” para agregar la imagen del menú/hamburguesa, un “H2” para el nombre de la hamburguesa que se vea mas grande, una “p” para la información de la hamburguesa, un “span” para el precio de la hamburguesa y el botón explicado anteriormente para agregar el producto al carrito.

```

<div class="grid-container">
  <div class="burger-card">
    
    <h2>Doble Carne</h2>
    <p>Una deliciosa hamburguesa con doble carne, queso, tomate y pepinillo</p>
    <span class="precio">7.90</span>
    <button class="botones_agregar" id="h1"> AGREGAR </button>
  </div>

```

Tenemos este “div” para el pie de pagina en el que esta la opción de la reserva y dentro de este “div” esta el “section” explicado anteriormente.

```

<div class="pie_de_pagina">
  <div class="img"></div>
  <div class="reserva">
    <div>
      <section id="reserva">
        <div class="container">
          <h2>Reserva tu Mesa</h2>
          <form>
            <label for="name">Nombre:</label>
            <input type="text" id="name" name="name" required>

            <label for="people">Número de Personas:</label>
            <input type="number" id="people" name="people" min="1" required>
          </form>
        </div>
      </section>
    </div>
  </div>

```

```

<label for="date">Fecha:</label>
<input type="date" id="date" name="date" required>

<label for="time">Hora:</label>
<input type="time" id="time" name="time" required>

<label for="phone">Teléfono:</label>
<input type="tel" id="phone" name="phone" required>

<label for="comments">Comentarios:</label>
<textarea id="comments" name="comments" rows="3"></textarea>

<button type="submit">Reservar</button>
</form>
</div>
</section>
</div>
</div>

```

(La imagen está dividida en 2, cada imagen es la continuidad de lo anterior)

## Otros

Al final del documento antes de cerrar el body tenemos el enlace al script para que podamos darle funcionalidad a todos los botones.

```
<script src="script.js"></script>
```

En el “head” tenemos:

El nombre de la página web, lo que te sale en la pestaña del navegador.

```
<title>Burger Factory</title>
```

El link para los estilos.

```
<link rel="stylesheet" href="styles.css">
```

# STYLES

En el “style”, simplemente hemos puesto todos los estilos para la visualización de la página y además hemos agregado el “@media” para que nuestra página sea responsiva, a continuación, dejare alguna imagen para la verificación de los hechos.

## *Sin @media*

```
body {
  font-family: sans-serif;
  margin: 0;
  background-color: black;
  color: white;
}

header {
  display: flex;
  min-height: 100px;
  background-color: rgb(35, 22, 22);
  justify-content: space-between;
  align-items: center;
  margin-bottom: 5px;
}

#cambiar_estilos{
  width: 30px;
  height: 30px;
  border-radius: 50%;
  margin-right: 10px;
}

.logo {
  display: flex;
}
```



## Con @media

```
@media (max-width: 700px) {  
  header {  
    flex-direction: column;  
  }  
  nav {  
    padding: 10px 0px;  
  }  
  nav a:hover {  
    color: white;  
  }  
}  
  
.botonCompra{  
  padding: 10px 20px;  
  background-color: #4CAF50;  
  color: white;  
  border: none;  
  border-radius: 5px;  
  cursor: pointer;  
}  
  
.botonCompra:active{  
  transform: scale(0.9);  
}
```

# SCRIPT

Voy a ir explicando uno por uno las cosas mas importantes de lo que hemos hecho en el script.

Esto son los enlazadores de los elementos de “html” para poder trabajar con ellos en “javascript”.

```
const bottonVisualizacion = document.getElementById("cambiar_estilos");
const botonesAgregar = document.querySelectorAll(".botones_agregar");
const carritoHtml = document.querySelector(".carrito_compra");
const carritoBoton = document.getElementById("carro_compra");
const total = document.getElementById("total");
let totalCompra = document.getElementById("carrito_total");
const body = document.querySelector('body');
const cabecera = document.querySelector('header');
const documento = document.querySelector('html');
const tarjetas = document.querySelectorAll(".burger-card");
```

## Funciones

Añadir funcionalidad y estilos al botón de mostrar carrito de la compra.

```
carritoBoton.addEventListener("click",revisarEstadoCarro);
carritoBoton.style.visibility = "hidden";
```

Función para que cuando pases el ratón por encima de un botón cambie de color

```
function cambiosDeColores(event){
    event.target.style.backgroundColor = "red";
    event.target.style.boxShadow = "2px 2px 2px darkred";
    event.target.innerHTML = "+";
    event.target.style.borderRadius = "50%";
}

function retornoColores(event){
    event.target.style.backgroundColor = "lightgray";
    event.target.style.boxShadow = "none";
    event.target.innerHTML = 'AGREGAR';
    event.target.style.borderRadius = "5px";
}
```

Función para manejar los datos de la hamburguesa que queremos agregar al carrito, y creamos un objeto con lo que nos hace falta de la hamburguesa.

```
function leerDatosHamburguesa(hamburguesa){
  let infoBurguer = {
    foto : hamburguesa.querySelector("img").src,
    precio : parseFloat(hamburguesa.querySelector("span").textContent),
    cantidad : 1,
    id: hamburguesa.querySelector("button").getAttribute("id")
  }
  //console.log(infoBurguer);
  almacenarObjetosIntermedios(infoBurguer);
  carritoBoton.style.visibility = "visible";
}
```

Función para agregar objetos almacenados a nuestro arreglo.

```
function almacenarObjetosIntermedios(infoBurguer){
  objetosAlmacenados.push(infoBurguer);
  //console.log(objetosAlmacenados);
}
```

Función del botón del carro de la compra que comprueba su estado.

```
function revisarEstadoCarro(event){
  event.preventDefault();
  if(objetosAlmacenados.length > 0){
    crearListadoCompra();
    event.target.style.backgroundColor = "lightgreen";
  }
}
```

Función que crea elementos en el carrito de la compra.

```
function crearListadoCompra(){
  carritoHtml.innerHTML = "";
  carritoHtml.style.opacity = "100%";
  carritoHtml.style.transitionDuration = "1.5s";
  carritoHtml.style.width = "fit-content";
  carritoHtml.style.height = "fit-content";
  totalCompra.innerHTML = "";
  let precioTotal = 0;
  objetosAlmacenados.forEach(objeto => {
    const espacioHamburguesa = document.createElement("div");
    const foto = document.createElement("img");
    foto.src = objeto.foto;
    foto.style.width = "100px";
    foto.style.height = "100px";
```

```

    espacioHamburguesa.appendChild(foto);
    const precio = document.createElement("span");
    precio.innerHTML = objeto.precio;
    precioTotal += objeto.precio;
    espacioHamburguesa.appendChild(precio);
    totalCompra.innerHTML = precioTotal;
    const cantidad = document.createElement("span");
    cantidad.innerHTML = objeto.cantidad;
    espacioHamburguesa.appendChild(cantidad);
    carritoHtml.appendChild(espacioHamburguesa);
    total.style.visibility = 'visible';
    totalCompra.style.visibility = 'visible';
    setTimeout(cerrarVentanaCarrito, 3000);
  })
}

```

(La imagen está dividida en 2, cada imagen es la continuidad de lo anterior)

Esta es la función para cambiar entre modo oscuro y modo claro. Lo que va haciendo es agregar una clase adicional a cada elemento de html y luego se codifica el css para esa clase en modo oscuro y modo claro. Así que cada elemento va a tener dos bloques en css, uno para oscuro y otro para claro.

```

bottonVisualizacion.addEventListener("click", ()=>{
  documento.classList.toggle("modo_claro");
  cabecera.classList.toggle('modo_claro');
  cambiaColores(cabecera);
  body.classList.toggle('modo_claro');
  cambiaColores(body);
  carritoHtml.classList.toggle('modo_claro');
  cambiaColores(carritoHtml);
  totalCompra.classList.toggle('modo_claro');
  total.classList.toggle('modo_claro');
  cambiaColores(totalCompra);
  cambiaColores(total);
});

```

Función para cambiar colores

```

function cambiaColores(elemento){
  if(elemento.classList.contains("modo_claro")){
    elemento.style.backgroundColor = estiloClaro.colorFondo;
    elemento.style.color = estiloClaro.color;
    elemento.style.boxShadow = estiloClaro.sombras;
    elemento.style.border = estiloClaro.bordes;
  }
}

```

```

    }else{
        elemento.style.backgroundColor = estiloOscuro.colorFondo;
        elemento.style.color = estiloOscuro.color;
        elemento.style.boxShadow = estiloOscuro.sombras;
        elemento.style.border = estiloOscuro.bordes;
    }
}

```

(La imagen está dividida en 2, cada imagen es la continuidad de lo anterior)

Función para el reloj de la pagina en cuenta regresiva para que el día 5 de marzo de 2025 a las 14:00 se acabe la cuenta regresiva

```

function cuentaRegresiva(flimite){
    let diferenciaFechas = flimite - Date.now();
    let segundosTotales = parseInt(diferenciaFechas / 1000);
    let segundos = parseInt(segundosTotales % 60);
    let minutosTotales = parseInt(segundosTotales / 60);
    let minutos = parseInt(minutosTotales % 60);
    let horasTotales = parseInt(minutosTotales / 60);
    let horas = parseInt(horasTotales % 24);
    let diasTotales = parseInt(horasTotales / 24);
    let cuentaRegresiva = diasTotales + " Dias " + horas + "Horas " + ":" + minutos + "Minutos " + ":" + segundos + "Segundos";
    mostrarCuentaAtras(cuentaRegresiva);
}

```

Función para desvanecer el mensaje y volver la visibilidad de la página a su estado normal y otra función llamada por un “setTimeout” para eliminar los elementos creados anteriormente.

```

function desvanecerMensaje(mensaje, parrafo){
    body.style.opacity = '100%';
    cabecera.style.opacity = '100%';
    cabecera.style.filter = 'none';
    body.style.filter = 'none';
    body.style.transitionDuration = '5s';
    cabecera.style.transitionDuration = '5s';
    mensaje.style.opacity = '0';
    mensaje.style.transitionDuration = '5s';
    setTimeout(eliminarMensaje, 1000, mensaje, parrafo);
}

```

Función que elimina definitivamente los elementos creados con anterioridad

```

function eliminarMensaje(mensaje, parrafo){
    mensaje.removeChild(parrafo);
    documento.removeChild(mensaje);
}

```