



# **Diseño de Interfaces Web**

GUÍA DE ESTILOS PROYECTO

Hecho por: Roy Lee Lunar Moré

**2º DAW**  
**Curso 2024-2025**  
**IES San Sebastián**



## INDICE

Memoria:.....	2
Descripción general del sitio.....	2
Aportación Extra:.....	3
Pruebas de Accesibilidad y Usabilidad.....	5
¿Qué partes del sitio web han requerido mas trabajo y por qué?.....	8
Aspectos a destacar.....	9
Diagrama de Actividad.....	10

## Memoria:

### Descripción general del sitio

#### ¿Qué es Iberic Burguers?

**R:** Iberic Burguers es una pequeña hamburguesería gourmet familiar que se ubica en Aracena, esta se encarga de combinar la producción Ibérica de la sierra con la comida rápida para la realización de una excelentes y deliciosas hamburguesas.

#### Publico Objetivo:

**R:** La hamburguesería tiene como objetivo dirigirse a un publico amplio combinando el diseño visual moderno y los elementos clásicos que representan a Aracena, la gente que accederá a la pagina serán fanáticos de la comida rápida, de los productos ibéricos y gente de Aracena.

Las intenciones son de que el mundo conozca el negocio y los alimentos qué esta realiza a través de Internet incitando a visitar el local y conocer sus hamburguesas.

#### Misión:

**R:** Su misión es darse a conocer en toda España para poder crecer y atraer publico a partir de la presentación de sus hamburguesas proveniente de productos 100% Ibéricos.

El logo de la empresa recopila partes importantes de lo que es la empresa en si, entre ellos destaca el icono principal que es una hamburguesa, la sierra de Aracena, y el cerdo como elementos secundarios y representativos de Aracena.

El logo se utiliza en la cabecera de la pagina como icono para representar el local, si se hace click en este nos redirigirá a la pagina de inicio, a su vez el logo se encuentra en el pie de pagina en la pagina desde la barra de paginas.

El tamaño del logo como se debe usar es de un ancho y largo de 72px x 72px.



## Aportación Extra:

**R:** La aportación que yo he decidido realizar es una sección con gráficos en forma de barra y líneas que permite visualizar de manera gráfica 2 tipos de datos: Uno son las ventas diarias y la otra son los productos mas vendidos en nuestra hamburguesería.

Estos gráficos les permiten tratar e interpretar a los visitantes de nuestro sitio web nuestras ventas para que puedan reconocer la fama de nuestros productos y nuestra hamburguesería. Estos gráficos se realizan mediante el uso de una librería llamada ChartJS.

**¿Qué es Chart.js?:** Chart.js es una librería de JavaScript que nos permite realizar gráficos sencillos que resultan ser dinámicos e interpretativos para los visitantes de tu sitio web. Es una librería muy ligera y sencilla de manejar, lo que es genial para representar datos de forma visual en cualquier aplicación web. Esta librería nos ofrece muchos tipos de gráficos como líneas, barras, circulares, entre otros además se nos permite personalizar colores, etiquetas y animaciones.

Para hacer uso de esta librería debemos crear un canvas HTML y definir un ID, luego en nuestro fichero JS importar Chart.JS y mediante el DOM el canvas HTML y crear un nuevo Chart donde especificamos el canvas y las opciones de nuestro Chart.

### **¿Qué gráficos he utilizado?**

**R:** He implementado 2 gráficos con la librería Chart.JS estos son los siguientes:

**Gráfico de líneas:** Este gráfico representa la evolución de las ventas a lo largo de la semana. En mi código se encuentra reflejado de la siguiente forma:

```
new Chart($("#ventasDiarias"), {
  type: "line",
  data: {
    labels: ["Lun", "Mar", "Mié", "Jue", "Vie", "Sáb", "Dom"],
    datasets: [
      {
        label: "Ventas",
        data: [90, 70, 120, 90, 130, 120, 90],
        borderColor: "#82898F",
      }
    ]
  },
  options: {
    responsive: true,
  }
});
```

En el esta definido el tipo del gráfico linear en este caso y asociamos una lista de datos con label en este caso son los días de la semana, luego asignamos los valores de ese datos en el mismo orden que definimos los labels, por ultimo definimos el color del borde y que sea responsive para que se adapte a los múltiples tamaños de pantalla.

El resultado final de los gráficos seria el siguiente:

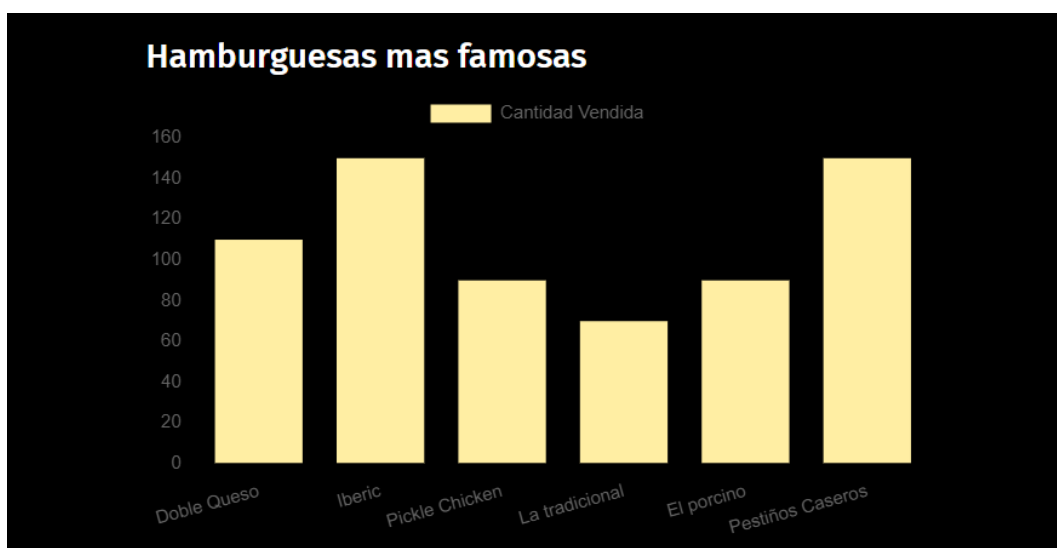


**Gráfico de barras:** Este gráfico representa el número de los productos mas vendidos en nuestro hamburguesería. Mi código para crear esta analítica es el siguiente:

```
new Chart($("#masVendidas"), {
  type: "bar",
  data: {
    labels: ["Doble Queso", "Iberic", "Pickle Chicken", "La tradicional", "El porcino", "Pestiños Caseros"],
    datasets: [
      {
        label: "Cantidad Vendida",
        data: [110, 150, 90, 70, 90, 150],
        backgroundColor: "#FFEEA3"
      }
    ]
  },
  options: {
    responsive: true,
  }
});
```

Su tipo de gráfico es de barra asociamos otra lista de datos que es un top 6 de los productos mas famosos, luego asignamos las valoraciones en orden con respecto a los productos y por ultimo definimos el color del fondo y que sea responsive.

El resultado final de los gráficos seria el siguiente:



## Pruebas de Accesibilidad y Usabilidad

La pagina sigue estándares de usabilidad y de accesibilidad como imágenes con texto alternativo para discernir el contenido de la imagen o video que esta tiene en caso de que alguien no pueda verlas, aria-label para especificar la acción que emitirá ese botón al hacer clic.

```
<!-- Main -->
<main class="container-fluid p-0">
  <div class="container-fluid bg-secondary p-5">
    <div class="container pt-5 pb-5">
      <div class="row d-flex flex-lg-row justify-content-center">
        <div class="col-md-4 d-flex align-items-center mb-4">
          <h2 data-translate="index_main-title">
            En nuestra hamburguesería de Aracena no solo servimos comida, creamos momentos inolvidables en familia.
          </h2>
        </div>
        <div class="col-md-4">
          
        </div>
      </div>
    </div>
  </div>
</div>
```

```
</div>
<p id="mensaje"></p>
<button type="button" id="btnEnviar" aria-label="Enviar Formulario"
  class="btn bg-secondary rounded-pill p-4 justify-self-center w-100" data-translate="valor-button="">>Enviar</button>
```

**Niveles de encabezado:** La pagina se encarga de cumplir a su vez con los niveles de encabezados para que las personas que usan lectores de pantalla no tengan problemas como perderse entre los títulos. Un buen contraste en la pagina para poder leer bien el texto y que no le dificulte a las personas la lectura, se siguen las leyes de Fitts para que el usuario no se le haga tan complicado hacer clic en un elemento. Se hace uso de roles y titulos en las etiquetas para identificar su acción o contenido y que los lectores de pantalla no tengan errores.

Se tiene una buena estructura en las etiquetas html para que no haya ningún error de sintaxis.

Los fondos poseen un buen contraste con la letra del fondo para cumplir con las pruebas de accesibilidad y el consorcio W3C.

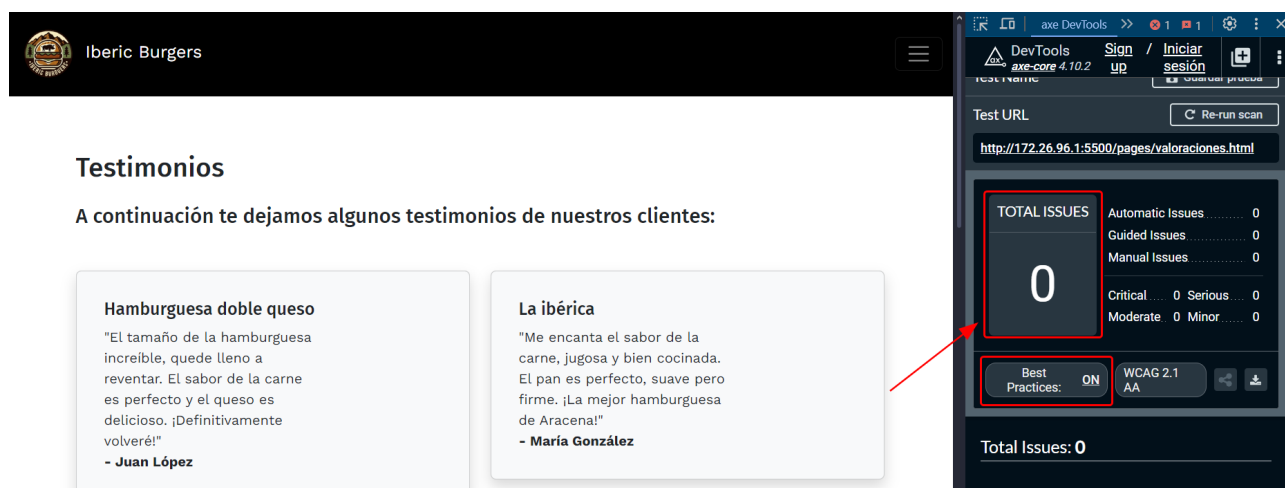


La pagina la pase por las pruebas de accesibilidad y usabilidad de la herramienta AxeDevTools para cubrir las pautas de accesibilidad de el consorcio W3C y que la pagina sea completamente accesible y usable.

Este es el resultado de la prueba en la pagina inicial. Como podremos ver en el apartado de problemas además de tener 0 errores cumplimos también con las mejores practicas aplicadas y la aplicación nos indica que también cumplimos con la Guía de Accesibilidad de Contenido de la W3C (WCAG 2.1) con un criterio AA.



Al igual que todas las demás paginas como las valoraciones cumplen con todas las pautas de accesibilidad.



The screenshot shows the 'Menu' page of the Iberic Burgers website. The page has a dark header with the logo and a main section with the text 'Menu' and 'Los mejores platillos de toda Aracena para degustar'. Below this is a table of products. The axe DevTools panel is open on the right, showing 'TOTAL ISSUES: 0' and 'Best Practices: ON'. Red arrows point from the table and the DevTools panel to the text below.

Producto	Precio	Categoría
Hamburguesa doble queso	12.50€	Hamburguesas

The screenshot shows the 'Reserva tu mesa' page of the Iberic Burgers website. The page has a dark header with the logo and a main section with the text 'Reserva tu mesa'. Below this is a form with fields for 'Nombre completo', 'Correo electrónico', 'Fecha de reserva', 'Hora de reserva', and 'Comentarios adicionales'. There is also a photo of the restaurant interior. The axe DevTools panel is open on the right, showing 'TOTAL ISSUES: 0' and 'Best Practices: ON'. Red arrows point from the form and the DevTools panel to the text below.

Como podremos ver todas las paginas cumplen con las pautas de accesibilidad universales.



## ¿Qué partes del sitio web han requerido mas trabajo y por qué?

**R:** La parte que más me ha costado trabajar ha sido el cambio del lenguaje ya que es algo que se tiene que aplicar a cada pagina con su traducción y adaptación correcta aun si agregas mas contenido.

El trabajo se volvió menos pesado gracias a JQuery que te facilita la manipulación del DOM, para traducir hago uso de un atributo personalizado llamado data-translate donde en el definiré el contenido del elemento que quiero traducir y luego con mi fichero JS me encargo de gestionar todo ahí:

```
<div class="container-fluid bg-secondary p-5">
  <div class="container pt-5 pb-5">
    <div class="row d-flex flex-lg-row justify-content-center">
      <div class="col-md-4 d-flex align-items-center mb-4">
        <h1 data-translate="index main-title" class="h2">
          En nuestra hamburguesería de Aracena no solo servimos comida, creamos momentos inolvidables
          en familia.
        </h1>
      </div>
      <div class="col-md-4">
        
        </div>
      </div>
    </div>
  </div>
</div>
```

El fichero recupera todos los elementos que tienen ese atributo y pasa un filtro para identificar elementos el contenido de ese elemento, si lo que quiero traducir tiene los siguientes atributos: "placeholder, aria-label, alt" cambiará el contenido encontrado ahí por la traducción si el elemento es un "optgroup" o un elemento normal cambiará su contenido también.

```
Codeium: Refactor | Explain | Generate JSDoc | X
function cambiarIdioma(idioma) {
  let elementos = document.querySelectorAll("[data-translate]");
  let language = $("html").attr("lang") == "es" ? "ingles" : "espanol";

  elementos.forEach((element) => {
    if (element.hasAttribute("alt")) {
      element.setAttribute("alt", idioma[language][element.getAttribute("data-translate")]);
    } else if (element.hasAttribute("placeholder")) {
      element.setAttribute("placeholder", idioma[language][element.getAttribute("data-translate")]);
    } else if (element.tagName === "OPTGROUP") {
      element.label = idioma[language][element.getAttribute("data-translate")];
    } else if (element.hasAttribute("aria-label")) {
      element.setAttribute("aria-label", idioma[language][element.getAttribute("data-translate")]);
    } else {
      element.innerHTML = idioma[language][element.getAttribute("data-translate")];
    }
  });

  $("html").attr("lang", $("html").attr("lang") == "es" ? "en" : "es");
  $("#buttonLang")
    .find("img")
    .attr("src", `src/icons/${$("html").attr("lang")}.png`);
  $("#changeLang")
    .find("img")
    .attr("src", `src/icons/${$("html").attr("lang") == "es" ? "en" : "es"}.png`);
  window.localStorage.setItem("lang", $("html").attr("lang") == "es" ? "en" : "es");
}
```

**¿Como hago la traducción?:** La traducción la hago extrayendo mediante AJAX la información de un fichero JSON en el están una gran mapa identificado por clave-valor donde la clave es el contenido del atributo data-translate, con esto identifico el contenido de dicho elemento y luego validando el elemento le agrego el contenido.

```
{
  "espanol": {
    "title-home": "Inicio | Iberic Burgers",
    "title-menu": "Menú | Iberic Burgers",
    "title-valoracion": "Valoraciones | Iberic Burgers",
    "title-reserve": "Reservar | Iberic Burgers",
    "title-aviso": "Aviso Legal | Iberic Burgers",
    "nav-title": "Navegación",
    "nav-link-start": "Inicio",
    "nav-link-menu": "Menú",
    "nav-link-valoracion": "Valoraciones",
    "nav-link-reserve": "Reservar",
    "nav-link-lang1": "Bandera de España",
    "nav-link-lang2": "Bandera de USA",
    "nav-link-span1": "Español",
    "nav-link-span2": "Inglés",
  }
}
```

Otra de las partes que me ha costado ha sido aprender el uso de AJAX mediante peticiones XMLHttpRequest para solicitar ficheros, una vez aprendido esto me permitió realizar la gran mayoría de mi proyecto porque uso muchos de estas solicitudes para los cambios de idioma DataTables y creación de contenido dinámico a través de información contenida en un fichero JSON.

## Aspectos a destacar

Puedo destacar que mi sitio web posee gráficos que permiten interpretar la información de forma rápida y efectiva tanto para trabajadores como para visitantes del sitio web.

- El **gráfico de líneas** muestra tendencias de ventas a lo largo de la semana.
- El **gráfico de barras** facilita la comparación entre los productos más vendidos.

Gracias a **Chart.js**, los gráficos son interactivos y fáciles de integrar en una web. Su presentación visual hace que cualquier persona, sin conocimientos técnicos, pueda entender los datos.

Mi página web posee un diseño atractivo y funcional, es moderno con una paleta de colores cálida que le brinda una sensación rústica al usuario. Posee un diseño mobile first lo que hace que la página se vea bien tanto en móvil, tableta o televisor como en un computador. A lo largo de todo el sitio web se utilizan múltiples librerías como lo son:

- JQuery
- DataTable
- JQuery Validator
- Node (Como gestor de paquetes)
- Chart.JS

Con estas librerías se cumplen todos los apartados solicitados en el proyecto como un DataTable con datos de productos, como un formulario validado con JQuery, gráficos bonitos con información importante y el uso constante de JQuery para la manipulación del DOM.

A su vez y por ultimo se incluyen una pagina de aviso legal para que quede mejor la página y el uso del gestor de paquetes Node.js para tener una buen estructura de ficheros y proyectos.

### Diagrama de Actividad

En este diagrama de Gantt podemos ver desde cuando comenzó y terminó cada apartado. Como se podrá ver en el apartado de abajo se ha estado trabajando constantemente en el proyecto y el apartado que mas se ha tardado en realizar es la traducción

