

# **T1 Ingeniería web I**

**Sebastian Perez Riaño**

**Universidad Manuela Beltrán**

**Ingeniería de software**

**Ingeniería web I**

**DIANA MARCELA TOQUICA**

## 1) Definiciones de términos:

- **Responsive:** Es una técnica de diseño web que permite que una página se adapte a diferentes tamaños de pantalla y dispositivos, como computadoras, tabletas y teléfonos móviles.
- **CSS Grid:** Es un sistema de diseño en CSS que permite organizar los elementos en filas y columnas facilitando la creación de diseños complejos y alineaciones necesarias. Todo esto ayuda a proporcionar un alto nivel de control sobre la distribución del contenido y permite crear estructuras responsivas de manera eficiente.
- **Flexbox:** Es un modelo de diseño en CSS que facilita la alineación y distribución de elementos en un contenedor que incluso cuando sus tamaños son dinámicos. Este es ideal para distribuir elementos en una sola dimensión, ya sea horizontal o vertical, y se usan combinados para organizar menús de navegación y estructuras flexibles.
- **Media queries:** Son reglas de CSS que permiten aplicar diferentes estilos según las características del dispositivo como el ancho de la pantalla. Se utilizan para hacer que un diseño sea responsivo y adaptarlo a distintos tamaños de pantalla.

## 2) Explicaciones importantes Código:

- **Un ejemplo del Grid:**

```
#productos { display: grid; grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));  
gap: 20px;  
justify-content: center;  
padding: 20px;  
}
```

En este caso se quería organizar los productos con un diseño cuadriculado con varias columnas, además, con la propiedad de “grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));” nos permite hacer un par de cosas más. El repeat(auto-fit, ...) nos permite indicar las columnas que se repetirán automáticamente según el espacio disponible. El minmax(250px, 1 fr) es el que define el tamaño de cada columna.

- **Un ejemplo del Flexbox:**

```
nav ul {  
  
  list-style: none;  
  
  display: flex;  
  
  justify-content: center;  
  
  gap: 20px;  
  
}
```

En este caso es ideal el Flexbox para distribuir elementos en una sola dimensión, con el display: flex; todos los elementos se colocan en una fila automáticamente, después se centran y se separan 20px.

Otro ejemplo sería #productos .producto ya que los elementos dentro de cada producto (imagen, texto y botón) se organizarían en columnas donde estarían centrados y el Flexbox lo que nos ayuda es a que el contenido del producto sea fácilmente escalable en diferentes tamaños de pantalla y se adapte sin desbordarse.

- **Ejemplo de media queries:**

```
@media (max-width: 768px) {
```

```
#productos {

grid-template-columns: 1fr; /* Una sola columna en pantallas pequeñas */

}

}
```

Aca se usa el media queries para optimizar el responsive en dispositivos móviles, este se activa cuando la pantalla es de 768px o menos, lo que hace es que queda una sola columna en vez de varias lo que mejora el como lo ve el usuario desde su pantalla móvil o Tablet.

### 3) Código:

#### HTML:

```
4) <!DOCTYPE html>
5) <html lang="es">
6) <head>
7)     <meta charset="UTF-8">
8)     <meta name="viewport" content="width=device-width, initial-scale=1.0">
9)     <title>Tienda de Bolsos</title>
10)    <link rel="stylesheet" href="../CSS/styles.css">
11) </head>
12) <body>
13)     <header>
14)         <h1>Tienda de Bolsos</h1>
15)         <nav>
16)             <ul>
17)                 <li><a href="../HTML/index.html">Inicio</a></li>
18)                 <li><a href="#">Productos</a></li>
19)                 <li><a href="#">Carrito</a></li>
20)             </ul>
21)         </nav>
22)     </header>
23)
24)     <main>
25)         <section id="productos">
26)             <h2>Lista de Productos</h2>
27)
28)             <div class="producto">
29)                 
30)                 <h3>Bolso Elegante</h3>
31)                 <p class="precio">$500.000</p>
```

```
32)         <button>Añadir al carrito</button>
33)     </div>
34)
35)     <div class="producto">
36)         
37)         <h3>Bolso de cuero</h3>
38)         <p class="precio">$250.000</p>
39)         <button>Añadir al carrito</button>
40)     </div>
41)
42)     <div class="producto">
43)         
44)         <h3>Bolso azul</h3>
45)         <p class="precio">$200.000</p>
46)         <button>Añadir al carrito</button>
47)     </div>
48)
49)     <div class="producto">
50)         
51)         <h3>Bolso rojo</h3>
52)         <p class="precio">$150.000</p>
53)         <button>Añadir al carrito</button>
54)     </div>
55)
56)     <div class="producto">
57)         
58)         <h3>Bolsos en promoción</h3>
59)         <p class="precio">$189.999</p>
60)         <button>Añadir al carrito</button>
61)     </div>
62) </section>
63)
64) <aside id="carrito">
65)     <h2>Carrito de Compras</h2>
66)     <div class="contenido">
67)         <p>Tu carrito está vacío</p>
68)     </div>
69) </aside>
70) </main>
71)
72) <footer>
73)     <p>&copy; 2025 Tienda de Bolsos</p>
74) </footer>
75) <script src="../scripts/script.js"></script>
76) </body>
```

```
77)</html>
```

## CSS:

```
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

body {
  font-family: Arial, Helvetica, sans-serif;
  background-color: #f5f5f5;
  color: #333;
  padding: 10px; /* Agrega un espacio para que el contenido no esté pegado */
}

header {
  background-color: #333;
  color: white;
  padding: 20px;
  text-align: center;
}

nav ul {
  list-style: none;
  display: flex;
  justify-content: center;
  gap: 20px;
}

nav ul li a {
  color: white;
  text-decoration: none;
}

button {
  background-color: bisque;
  color: #333;
}

/* Diseño de Productos */
#productos {
  display: grid;
```

```
    grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));
    gap: 20px;
    justify-content: center; /*Cambios video*/
    padding: 20px;
    max-width: 1000px;
    margin: auto;
}

#productos .producto {
    background-color: white;
    padding: 15px;
    border-radius: 10px;
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
    text-align: center;
    display: flex;
    flex-direction: column;
    align-items: center;
    transition: transform 0.3s ease-in-out;
}

#productos .producto:hover {
    transform: scale(1.05);
}

.producto img {
    width: 180px;
    height: auto;
    border-radius: 8px;
    margin-bottom: 10px;
}

.producto p {
    font-size: 16px;
    color: #555;
    margin-bottom: 5px;
}

.producto .precio {
    font-size: 18px;
    font-weight: bold;
    color: #333;
    margin-bottom: 10px;
}

.producto button {
```

```
background-color: #ff6600;
color: white;
border: none;
padding: 10px 15px;
font-size: 16px;
cursor: pointer;
border-radius: 5px;
transition: background-color 0.3s;
width: 100%; /* Hace que el botón ocupe todo el ancho */
}

.producto button:hover {
    background-color: #cc5500;
}

/* Carrito */
#carrito {
    background-color: white;
    padding: 15px;
    margin: 20px auto;
    border-radius: 5px;
    text-align: center;
    max-width: 400px; /* Ajusta el ancho */
    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);
}

.producto-carrito {
    display: flex;
    justify-content: space-between;
    align-items: center;
    background: #fff;
    padding: 10px;
    margin: 5px 0;
    border-radius: 5px;
    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);
}

.producto-carrito .eliminar {
    background: red;
    color: white;
    border: none;
    padding: 5px 10px;
    cursor: pointer;
    border-radius: 5px;
}
```



```

.producto-carrito .eliminar:hover {
    background: darkred;
}

/* Estilos Responsive */
@media (max-width: 768px) {
    #productos {
        grid-template-columns: 1fr;
    }

    .producto img {
        width: 100px; /* Reduce el tamaño en móviles */
    }

    .producto button {
        font-size: 14px; /* Botón más pequeño en pantallas chicas */
    }
}

```

### Scripts.js:

```

document.addEventListener("DOMContentLoaded", () => {
    const botonesAgregar = document.querySelectorAll(".producto button"); //
    Captura todos los botones de añadir
    const carrito = document.querySelector("#carrito"); // Captura la sección del
    carrito
    let carritoItems = []; // Lista de productos en el carrito

    // Agregar evento a cada botón de "Añadir al carrito"
    botonesAgregar.forEach((boton) => {
        boton.addEventListener("click", (event) => {
            const producto = event.target.parentElement; // Captura el producto
            contenedor
            const nombre = producto.querySelector("h3").textContent; // Nombre
            del producto
            const precio = producto.querySelector(".precio").textContent; //
            Precio del producto

            // Agregar producto al carrito
            carritoItems.push({ nombre, precio });
            actualizarCarrito();
        });
    });
}

```

```

});

// Función para actualizar el carrito
function actualizarCarrito() {
    const carritoContenido = document.querySelector("#carrito .contenido");
    if (!carritoContenido) {
        const div = document.createElement("div");
        div.classList.add("contenido");
        carrito.appendChild(div);
    } else {
        carritoContenido.innerHTML = ""; // Vaciar contenido antes de
actualizar
    }

    if (carritoItems.length === 0) {
        carritoContenido.innerHTML = "<p>Tu carrito está vacío</p>";
    } else {
        carritoItems.forEach((item, index) => {
            const productoCarrito = document.createElement("div");
            productoCarrito.classList.add("producto-carrito");
            productoCarrito.innerHTML = `
                <p>${item.nombre} - ${item.precio}</p>
                <button class="eliminar" data-index="${index}">X</button>
            `;
            carritoContenido.appendChild(productoCarrito);
        });

        // Agregar eventos a los botones de eliminar
        document.querySelectorAll(".eliminar").forEach((boton) => {
            boton.addEventListener("click", (event) => {
                const index = event.target.getAttribute("data-index");
                carritoItems.splice(index, 1); // Eliminar producto del
carrito

                actualizarCarrito(); // Actualizar la vista del carrito
            });
        });
    }
}

});

```

#### 4) Imágenes del Responsive:

##### Imagen 1)

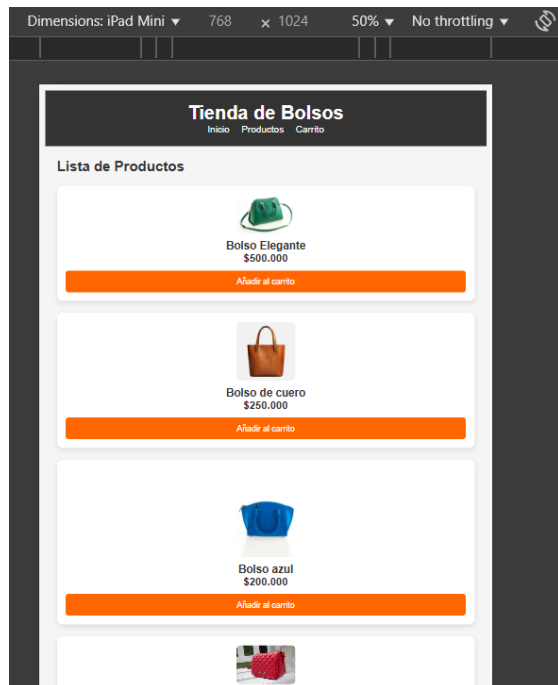
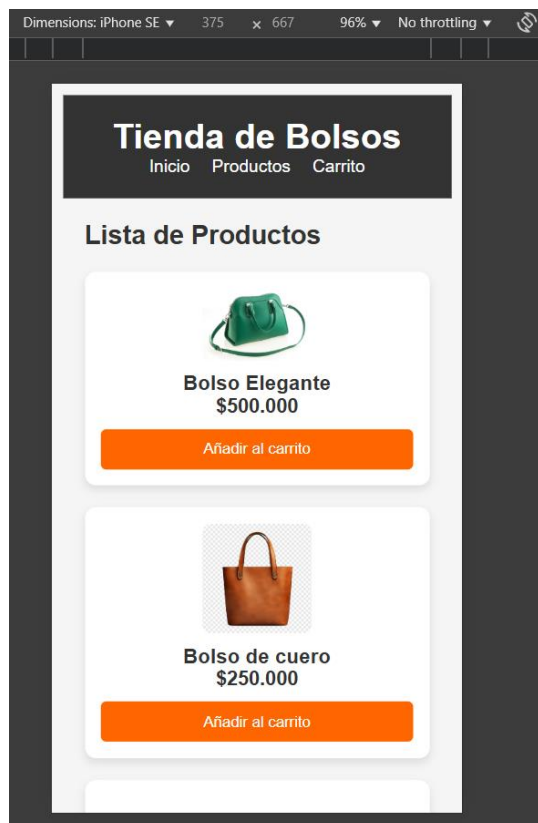


Imagen 2)



5)