

Para que el menú de tu página de **WebSur Studio** se **contraiga en dispositivos móviles** y se muestre únicamente al pulsar un “ícono hamburguesa”, necesitas implementar un comportamiento que:

1. **Cambie la disposición (layout) del menú** en pantallas pequeñas, haciendo que inicialmente no aparezcan los enlaces o aparezcan ocultos.
2. **Incluya un botón** (generalmente el ícono de “tres líneas” o hamburguesa) que, al pulsarlo, despliegue u oculte el menú.

Esto puede hacerse de varias maneras. A continuación, tienes la **idea general** con HTML, CSS y un poquito de JavaScript que te permiten lograrlo de forma sencilla:

## 1. ESTRUCTURA HTML SUGERIDA

Dentro de tu <header> o estructura principal, puedes cambiar la parte del menú para que tenga un botón con clase .menu-toggle (el ícono hamburguesa) y un contenedor que agrupe tus enlaces:

```
<header class="header">
  <div class="container header-content">

    <!-- Logo -->
    <a href="index.html" class="logo">
      
    </a>

    <!-- Botón Hamburguesa (visible solo en móvil) -->
    <button class="menu-toggle" aria-label="Abrir menú">
      <!-- Puedes usar un ícono o un svg de hamburguesa -->
      ≡
    </button>

    <!-- Navegación principal -->
    <nav class="main-nav">
      <ul>
        <li><a href="index.html">Inicio</a></li>
        <li><a href="about.html">Sobre Nosotros</a></li>
        <li><a href="services.html">Servicios</a></li>
        <li><a href="portfolio.html">Portafolio</a></li>
        <li><a href="team.html">Equipo</a></li>
        <li><a href="contact.html">Contacto</a></li>
      </ul>
    </nav>
  </div>
</header>
```

```
        </ul>
      </nav>

    </div>
  </header>
```

#### Puntos clave:

- **.menu-toggle**: El botón que se mostrará como icono de menú cuando la pantalla sea pequeña.
- **.main-nav**: El contenedor de los enlaces de navegación. Se mostrará/ocultará en función de la acción del usuario y el tamaño de la pantalla.

## 2. CSS PARA MOSTRAR/OCULTAR MENÚ EN DISPOSITIVOS MÓVILES

La idea es:

- **Por defecto**, en pantallas pequeñas, ocultar los enlaces y mostrar el botón hamburguesa.
- En pantallas grandes, mostrar el menú normal y ocultar el botón hamburguesa.

Un ejemplo de reglas responsivas podría ser:

```
/* 1. Layout básico del header */
.header {
  background: #fff;
  padding: 1rem 0;
  position: fixed; /* Si lo quieres fijo arriba */
  top: 0;
  width: 100%;
  z-index: 999;
}

.header-content {
  display: flex;
  align-items: center;
  justify-content: space-between;
  width: 90%;
  max-width: 1200px;
  margin: 0 auto;
}
```

```
/* 2. Estilos generales del nav */
.main-nav ul {
  list-style: none;
  margin: 0;
  padding: 0;
  display: flex; /* Por defecto, los enlaces en línea */
  gap: 1rem;
}

/* 3. Botón hamburguesa (por defecto oculto en escritorio) */
.menu-toggle {
  display: none;
  background: none;
  border: none;
  font-size: 1.5rem;
  cursor: pointer;
}

/* =====
Reglas para móviles
(por ejemplo, pantallas <= 768px)
===== */
@media (max-width: 768px) {
  /* Ocultar el menú de enlaces por defecto en móviles */
  .main-nav {
    position: absolute;
    top: 60px; /* Ajusta según la altura del header */
    right: 0;
    background: #fff;
    width: 200px;
    box-shadow: 0 0 8px rgba(0,0,0,0.1);

    /* Por defecto, vamos a ocultarlo con display: none */
    display: none;
  }
}
```

```
flex-direction: column;
gap: 0; /* Elimina el gap para acomodar vertical */
}

.main-nav ul {
  flex-direction: column;
  margin: 1rem 0;
}

.main-nav li {
  margin-bottom: 0.5rem;
  text-align: center;
}

/* Mostrar el botón hamburguesa en móvil */
.menu-toggle {
  display: block;
}

/*
  Clase .open que se aplicará
  al nav cuando el usuario haga clic en el botón
*/
.main-nav.open {
  display: flex; /* Mostramos el nav cuando esté la clase .open */
}
}
```

#### Explicación:

1. **.menu-toggle { display: none; }** inicialmente en escritorio, para que no se vea si la pantalla es ancha.
  2. **.menu-toggle** se vuelve visible en media queries `max-width: 768px;` mientras tanto, se oculta el menú real (`.main-nav`) poniendo `display: none;`.
  3. Cuando el usuario haga clic en el botón, se le añade una clase (por ejemplo, `.open`) a `.main-nav` para que pase a `display: flex;`.
-

### 3. JAVASCRIPT PARA ACTIVAR EL MENÚ DESPLEGABLE

Finalmente, necesitas un pequeño script para **escuchar el evento de clic** en el botón `.menu-toggle` y alternar la clase `.open` en `.main-nav`. Por ejemplo, al final del `<body>`, antes de cerrar la etiqueta, puedes poner:

```
<script>

// Seleccionamos el botón y el nav
const toggleBtn = document.querySelector('.menu-toggle');
const nav = document.querySelector('.main-nav');

toggleBtn.addEventListener('click', () => {
  // Al hacer clic, alternamos la clase .open
  nav.classList.toggle('open');
});
</script>
```

**Función `classList.toggle('open')`:**

- Si `.main-nav` NO tiene la clase `.open`, se la añade, lo que hace `display: flex`; (muestra el menú).
- Si sí la tiene, se la quita, volviendo a `display: none`;

---

### 4. RESUMEN DEL FLUJO EN DISPOSITIVOS MÓVILES

1. **Por defecto**, en pantallas pequeñas (`@media (max-width: 768px)`), el menú (`.main-nav`) tiene `display: none`; y se muestra el botón hamburguesa (`.menu-toggle`).
2. El usuario hace clic en el botón hamburguesa.
3. El JavaScript añade la clase `.open` al `<nav class="main-nav">`, que cambia `display: none` a `display: flex`;
4. Los enlaces se muestran en un menú desplegable (vertical), hasta que el usuario vuelva a pulsar el botón o cambie de vista.
5. En pantallas anchas (por encima de `768px`), el menú vuelve a su forma horizontal, el botón hamburguesa se oculta (`display: none`); y `.main-nav` se muestra siempre (`display: flex`;) sin necesidad de JavaScript para el despliegue.

---

### CONSIDERACIONES ADICIONALES

- Ajusta los valores de `max-width: 768px` u otras medidas a tus necesidades.
- Puedes usar un ícono de hamburguesa (un SVG, un `<span>` con `background`, etc.) en vez de `≡`, para mejorar la apariencia.
- Para que el header fijo no tape el contenido, debes añadir un `margin-top` apropiado al `body` o al primer `<main>` cuando el header está `position: fixed`;