

# **PÁGINA WEB “STEARWAY-NORTH POLE”**

## **MANUAL TECNICO**

### **1. INTRODUCCIÓN**

Este documento describe la estructura técnica de la página web “StearWay-North Pole”, incluyendo tecnologías empleadas, organización del código y funciones implementadas.

### **2. TECNOLOGÍAS UTILIZADAS**

En Visual Studio Code, creamos 3 archivos:

- **HTML:** Es la estructura de la página.
- **CSS:** Estilos y diseño responsivo de la página.
- **JAVASCRIPT:** Lógica del menú y bloqueo de navegación si no se inicia sesión.

### **3. DESCRIPCIÓN DE FUNCIONAMIENTO**

#### **Responsividad**

El CSS contiene Media Queries para:

- Tablets (hasta 1024px)
- Celulares (hasta 550px)

Los productos pasan de 4 columnas en laptop, 2 columnas para tablet y 1 columna para celulares

En celulares el menú se convierte en menú hamburguesa.

#### **Sistema de Inicio de Sesión**

- Se almacena el acceso con:

```
if (correo === correoGuardado && contra === contraGuardada) {  
    localStorage.setItem("sesionActiva", "true");  
    alert("BIENVENIDO A STEARWAY NORTH POLE");  
    window.location.href = "menu.html";  
}
```

- Si el usuario intenta entrar a otra página sin iniciar sesión:

Se muestra una alerta si quiere entrar a otra sección sin iniciar sesión y se lo redirige a inicio.html

```
if (sesion !== "true" && !window.location.href.includes("index.html"))  
    alert("Debes registrarte primero");
```

Se muestra una alerta si los campos de iniciar sesión no se llenaron

```
if (correo === "" || contra === "") {  
    alert("Completa todos los campos");  
    return;  
}
```

## Menú en Móvil

El menú hamburguesa ejecuta una función para mostrar u ocultar opciones:

```
function toggleMenu() {  
    const nav = document.querySelector("nav");  
    nav.classList.toggle("ventanas");  
}
```

Esto cambia la visibilidad del menú usando CSS.

## 5. GLOSARIO TÉCNICO

- **display:none:** Oculta un elemento completamente, eliminándolo del flujo del documento y haciendo que no ocupe espacio, como si no existiera, permitiendo que otros elementos ocupen su lugar.

- **display:flex:** Convierte un elemento en un contenedor flexible
- **Pointer-event: none:** hace que un elemento HTML ignore por completo los eventos del ratón
- **list-style:none:** es una propiedad abreviada que se usa para quitar las viñetas de los elementos de una lista
- **display: inline-block:** Es un valor que permite que un elemento se comporte como un elemento en línea, colocándose uno al lado del otro, pero a la vez acepta propiedades de caja de un elemento de bloque
- **text-decoration: none:** Es una propiedad abreviada que elimina cualquier línea decorativa del texto, como el subrayado predeterminado de los enlaces
- **flex-wrap: wrap:** Es una propiedad de Flexbox que permite a los elementos flexibles pasar a la siguiente línea si no caben en una sola fila dentro de su contenedor, en lugar de encogerse.
- **Solid:** Para la propiedad border-style creando un borde liso y continuo
- **RGBA:** Significa Rojo, Verde, Azul, Alfa (opacidad)
- **display: block:** convierte un elemento HTML en un bloque, haciendo que ocupe el 100% del ancho disponible
- **position: absolute:** hace que un elemento se coloque respecto a su contenedor posicionado más cercano.
- **flex-direction: column:** Es una propiedad del contenedor flex que cambia el eje principal de horizontal a vertical
- **text-decoration: underline:** Es una propiedad para subrayar texto, dibujando una línea debajo de él
- **flex: nowrap:** Es un valor para la propiedad flex-wrap que le dice a los elementos flexibles que no se ajusten a nuevas líneas, se mantendrán en una sola fila
- **localStorage:** Es una API del navegador para almacenar datos de forma persistente localmente en el dispositivo del usuario, sin fecha de expiración
- **e.preventDefault():** Método que impide la acción por defecto de un evento en el navegador.
- **getItem:** toma una clave y devuelve el valor correspondiente

- **setItem:** se usa para guardar datos en el navegador del usuario (LocalStorage)
- **Json.Parse:** es un método fundamental que toma una cadena de texto en formato JSON y la convierte en un objeto JavaScript.
- **Push:** Es un método de los que permite agregar uno o más elementos al final del original.
- **JSON.stringify:** Método que convierte un objeto o valor de JavaScript en una cadena de texto con formato JSON
- **JSON:** formato ligero y legible por humanos para el intercambio de datos estructurados, que utiliza clave , valor y arreglos.