

```

// --- Lógica para el Menú Móvil ---
const menuBtn = document.getElementById('menu-btn');
const mobileMenu = document.getElementById('mobile-menu');

// Alterna la visibilidad del menú al hacer clic en el botón
menuBtn.addEventListener('click', () => {
  mobileMenu.classList.toggle('hidden');
});

// Cierra el menú móvil cuando se hace clic en un enlace
mobileMenu.addEventListener('click', (e) => {
  if (e.target.tagName === 'A') {
    mobileMenu.classList.add('hidden');
  }
});

// --- Lógica para el envío del Formulario de Contacto (usando
// Formspree) ---
const form = document.getElementById('contact-form');
const statusDiv = document.getElementById('form-status');

// Función asíncrona para manejar el envío del formulario
async function handleSubmit(event) {
  event.preventDefault(); // Evita que la página se recargue
  const data = new FormData(event.target);

  // Muestra un mensaje de estado mientras se envía
  statusDiv.innerHTML = 'Enviando...';
  statusDiv.classList.remove('text-green-600', 'text-red-600');

  // Envía los datos al servicio de Formspree
  fetch(event.target.action, {
    method: form.method,
    body: data,
    headers: {
      'Accept': 'application/json'
    }
  }).then(response => {
    // Si el envío fue exitoso
    if (response.ok) {
      statusDiv.innerHTML = "¡Gracias por tu mensaje! Te
contactaré pronto.";
      statusDiv.classList.add('text-green-600');
      form.reset(); // Limpia el formulario
    } else {
      // Si hubo un error en la respuesta
      response.json().then(data => {
        if (Object.hasOwn(data, 'errors')) {

```

```

        statusDiv.innerHTML = data["errors"].map(error =>
error["message"]).join(", ");
    } else {
        statusDiv.innerHTML = "Oops! Hubo un problema al
enviar tu formulario.";
    }
    statusDiv.classList.add('text-red-600');
    })
    }
    }).catch(error => {
        // Si hubo un error en la conexión
        statusDiv.innerHTML = "Oops! Hubo un problema al enviar tu
formulario.";
        statusDiv.classList.add('text-red-600');
    });
}

// Asigna la función 'handleSubmit' al evento 'submit' del formulario
form.addEventListener("submit", handleSubmit);

// --- Inicializa los Iconos de Lucide ---
// Esta función busca todos los elementos con el atributo
'data-lucide'
// y los reemplaza con el SVG del icono correspondiente.
lucide.createIcons();

```