**⚙️ 2. Instalar Node.js y Express**

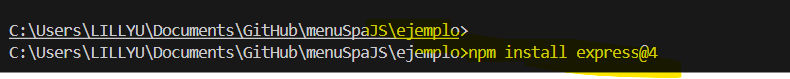
Primero asegúrate de tener Node.js instalado. Luego:

mkdir mi-portafolio-spa

cd mi-portafolio-spa

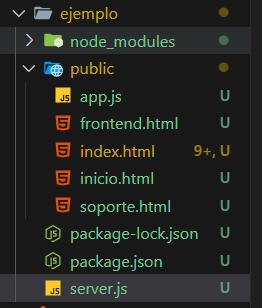
npm init -y

npm install express@4



|  |  |
| --- | --- |
| Comando | Explicación breve |
| mkdir mi-portafolio-spa | Crea una nueva carpeta para tu proyecto |
| cd mi-portafolio-spa | Entra a esa carpeta |
| npm init -y | Crea el archivo package.json automáticamente con valores por defecto |
| npm install express@4 | Instala la versión **estable** de Express (v4.x) en el proyecto |

Carpetas como quedan , debes crear los archivos y carpetas necesarias.



**📁 3. Crea el servidor Node.js (server.js)**

// server.js

const express = require('express');// Importa el módulo express

// Importa el módulo path para manejar rutas de archivos

const path = require('path'); // Importa el módulo path para manejar rutas de archivos

// Crea una instancia de la aplicación Express

const app = express();// Crea una instancia de la aplicación Express

// Define el puerto en el que el servidor escuchará

const PORT = 3000;// Define el puerto en el que el servidor escuchará

// Ruta estática

app.use(express.static(path.join(\_\_dirname, 'public')));// Define la carpeta 'public' como la ruta estática

// Cualquier ruta no capturada (SPA fallback)

app.get('/\*', (req, res) => {// Cualquier ruta no capturada (SPA fallback)

  res.sendFile(path.join(\_\_dirname, 'public', 'index.html'));// Envía el archivo index.html para cualquier ruta no capturada

});

app.listen(PORT, () => {/// Inicia el servidor en el puerto definido

  console.log(`Servidor corriendo en http://localhost:${PORT}`);// Inicia el servidor en el puerto definido

});

**📜 4. Lógica SPA (public/app.js)**

// Función para cargar secciones HTML en el contenedor principal

async function loadSection(page) {// Función para cargar secciones HTML en el contenedor principal

  const container = document.getElementById('content-container'); // Contenedor principal

  try {

    const response = await fetch(page); // Hace la petición del archivo

    if (!response.ok) throw new Error('Página no encontrada'); // Error si falla

    const html = await response.text(); // Convierte a texto

    container.innerHTML = html; // Inserta el contenido

  } catch (error) {

    container.innerHTML = `<h2 class="text-red-600">Error:</h2><p>${error.message}</p>`; // Muestra el error

  }

}

// Cargar la sección de inicio por defecto al cargar la página

document.addEventListener('DOMContentLoaded', () => {// Cargar la sección de inicio por defecto al cargar la página

  loadSection('inicio.html');

});

// Alternar el menú en dispositivos móviles

document.getElementById('menu-toggle')?.addEventListener('click', () => {// Alternar el menú en dispositivos móviles

  document.getElementById('menu')?.classList.toggle('active');// Alterna la clase 'active' del menú

});// Alternar el menú en dispositivos móviles

**💡 5. Crea páginas individuales**

Ejemplo de inicio.html:

<h2 class="text-3xl font-bold text-indigo-700 mb-4">Inicio</h2>

<p class="text-gray-800 leading-relaxed">

¡Bienvenido a mi portafolio! Aquí encontrarás toda la información sobre mis servicios, análisis de mercado y documentación del proyecto.

</p>

Crea los demás archivos con contenido similar y nombrados como:

* frontend.html
* soporte.html
* diseno.html
* …etc. (usa la lista que ya te di antes)

**🚀 6. Ejecutar la SPA**

Desde tu terminal:

node server.js

Y abre tu navegador en:  
👉 [**http://localhost:3000**](http://localhost:3000)

1. **Pega el siguiente código** en index.html:
2. <!DOCTYPE html>
3. <html lang="es">
4. <head>
5. <meta charset="UTF-8" />
6. <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
7. <title>Portafolio SPA con Node.js</title>
8. <script src="https://cdn.tailwindcss.com"></script>
9. <style>
10. body {
11. overflow-x: hidden; /\* Evita el scroll horizontal al abrir el menú \*/
12. display: flex; /\* Añadido para layout con footer pegado abajo \*/
13. flex-direction: column; /\* Contenido en columna para el footer \*/
14. min-height: 100vh; /\* Asegura que el body ocupe al menos toda la altura de la ventana \*/
15. }
16. #menu {
17. position: fixed;
18. top: 60px; /\* Debajo del encabezado fijo \*/
19. left: 0;
20. height: calc(100% - 60px);
21. width: 250px;
22. background-color: white;
23. padding: 1rem;
24. box-shadow: 2px 0 5px rgba(0, 0, 0, 0.1);
25. transform: translateX(-100%); /\* Oculto por defecto en móviles \*/
26. transition: transform 0.3s ease-in-out;
27. z-index: 20; /\* Asegura que el menú esté por encima del contenido \*/
28. overflow-y: auto; /\* Permite scroll si el contenido del menú es largo \*/
29. }
30. #menu.active {
31. transform: translateX(0); /\* Visible \*/
32. }
33. .main-content-area { /\* Contenedor para el contenido dinámico \*/
34. margin-left: 0; /\* Sin margen por defecto \*/
35. transition: margin-left 0.3s ease-in-out;
36. padding-top: 80px; /\* Espacio para el encabezado fijo \*/
37. flex-grow: 1; /\* Permite que el contenido ocupe el espacio disponible y empuje el footer \*/
38. }
39. @media (min-width: 768px) {
40. #menu {
41. transform: translateX(0); /\* Siempre visible en desktop \*/
42. position: fixed; /\* Asegura que siga siendo flotante en desktop \*/
43. }
44. .main-content-area {
45. margin-left: 250px; /\* Empuja el contenido cuando el menú es visible \*/
46. }
47. #menu-toggle {
48. display: none; /\* Oculta el botón de alternancia en desktop \*/
49. }
50. }
51. .footer {
52. background-color: #333;
53. color: white;
54. text-align: center;
55. padding: 1rem;
56. width: 100%;
57. box-sizing: border-box; /\* Incluye padding en el ancho \*/
58. /\* Posicionamiento en mobile y desktop para estar al final \*/
59. margin-top: auto; /\* Empuja el footer hacia abajo si el contenido no llena la pantalla \*/
60. }
61. @media (min-width: 768px) {
62. .footer {
63. margin-left: 250px; /\* Empuja el footer en desktop también \*/
64. width: calc(100% - 250px);
65. }
66. }
67. </style>
68. </head>
69. <body class="bg-gray-100">
70. <div class="p-4 bg-indigo-600 text-white flex items-center justify-between fixed w-full z-30">
71. <div class="flex items-center gap-3">
72. <img src="https://via.placeholder.com/40" alt="Logo" class="w-10 h-10 rounded-full">
73. <span class="text-xl font-semibold">Mi Portafolio</span>
74. </div>
75. <button id="menu-toggle" class="md:hidden">
76. <svg class="w-7 h-7 text-white" fill="none" stroke="currentColor" viewBox="0 0 24 24" xmlns="http://www.w3.org/2000/svg">
77. <path stroke-linecap="round" stroke-linejoin="round" stroke-width="2" d="M4 6h16M4 12h16M4 18h16"></path>
78. </svg>
79. </button>
80. </div>
81. <nav id="menu" class="shadow-lg">
82. <ul class="space-y-4">
83. <li>
84. <button onclick="loadSection('inicio.html')" class="w-full text-left font-medium hover:text-indigo-600">
85. 🏠 Inicio
86. </button>
87. </li>
88. <li class="group">
89. <details>
90. <summary class="cursor-pointer font-medium hover:text-indigo-600">💼 Servicios (Portafolio)</summary>
91. <ul class="pl-4 mt-2 space-y-1 text-sm text-gray-700">
92. <li><button onclick="loadSection('frontend.html')">↳ Desarrollo Frontend</button></li>
93. <li><button onclick="loadSection('soporte.html')">↳ Soporte Técnico</button></li>
94. <li><button onclick="loadSection('diseno.html')">↳ Diseño UX/UI</button></li>
95. </ul>
96. </details>
97. </li>
98. <li class="group">
99. <details>
100. <summary class="cursor-pointer font-medium hover:text-indigo-600">📊 Análisis de Mercado</summary>
101. <ul class="pl-4 mt-2 space-y-1 text-sm text-gray-700">
102. <li><button onclick="loadSection('problema.html')">↳ Problema detectado</button></li>
103. <li><button onclick="loadSection('publico.html')">↳ Público Objetivo</button></li>
104. <li><button onclick="loadSection('competencia.html')">↳ Competencia</button></li>
105. </ul>
106. </details>
107. </li>
108. <li class="group">
109. <details>
110. <summary class="cursor-pointer font-medium hover:text-indigo-600">💰 Cotización del Proyecto</summary>
111. <ul class="pl-4 mt-2 space-y-1 text-sm text-gray-700">
112. <li><button onclick="loadSection('sueldos.html')">↳ Sueldos por rol</button></li>
113. <li><button onclick="loadSection('hosting.html')">↳ Costos de dominio y hosting</button></li>
114. <li><button onclick="loadSection('valor.html')">↳ Valor final estimado</button></li>
115. </ul>
116. </details>
117. </li>
118. <li class="group">
119. <details>
120. <summary class="cursor-pointer font-medium hover:text-indigo-600">📄 Documentación Técnica</summary>
121. <ul class="pl-4 mt-2 space-y-1 text-sm text-gray-700">
122. <li><button onclick="loadSection('funcionales.html')">↳ Requerimientos Funcionales</button></li>
123. <li><button onclick="loadSection('no-funcionales.html')">↳ No Funcionales</button></li>
124. <li><button onclick="loadSection('problemas.html')">↳ Problemas y Soluciones</button></li>
125. <li><button onclick="loadSection('scrum.html')">↳ Roles y SCRUM</button></li>
126. </ul>
127. </details>
128. </li>
129. <li class="group">
130. <details>
131. <summary class="cursor-pointer font-medium hover:text-indigo-600">📞 Contacto</summary>
132. <ul class="pl-4 mt-2 space-y-1 text-sm text-gray-700">
133. <li><button onclick="loadSection('formulario.html')">↳ Formulario de contacto</button></li>
134. <li><button onclick="loadSection('redes.html')">↳ Redes sociales (simuladas)</button></li>
135. </ul>
136. </details>
137. </li>
138. </ul>
139. </nav>
140. <main id="content-container" class="main-content-area p-6">
141. <h2 class="text-2xl font-bold text-indigo-600 mb-2">Cargando...</h2>
142. <p class="text-gray-700">Por favor, espera mientras se carga el contenido.</p>
143. </main>
144. <footer class="footer">
145. <p>&copy; 2025 Mi Portafolio. Todos los derechos reservados.</p>
146. </footer>
147. <script src="app.js" defer></script>
148. </body>
149. </html>

Crea cada pagina solicitada par el proyecto :

frontend.html

soporte.html

diseno.html

problema.html

publico.html

competencia.html

sueldos.html

hosting.html

valor.html

funcionales.html

no\_funcionales.html

problemas.html

scrum.html

formulario.html

redes.html

inicio.html