

Contención Visual y Desbordamiento en Bloques de Código

Estantería: Desarrollo y Productividad

Subtema: Estilos

mercedev.es — 2026-06-13 | Epic 8 - Fase 6

El Desafío (Síntoma)

Durante las pruebas de calidad (QA) de la Biblioteca de cuadernillos, se detectó que los fragmentos de código, envueltos en las etiquetas semánticas `<pre>` y `<code>`, no respetaban los límites del contenedor principal (`viewport`).

Esto provocaba un doble problema crítico: 1. **En dispositivos móviles:** El texto preformateado estiraba el ancho total de la página de forma artificial, forzando la aparición de un incómodo *scroll* horizontal global, lo que penalizaba las métricas de usabilidad de Core Web Vitals (CLS y Mobile Friendliness). 2. **En la exportación documental:** Al utilizar motores de renderizado como WeasyPrint para generar las versiones PDF del ecosistema, las líneas de código largas se truncaban, ocultando información vital fuera de los márgenes del papel.

La Maniobra (Lógica)

Se aplicó un parche a nivel arquitectónico en la hoja de estilos base (`src/scss/base/_typography.scss`), estableciendo reglas de contención rígidas e implacables sobre ambas etiquetas semánticas:

Se inyectaron las directivas `white-space: pre-wrap;` y `word-break: break-word;`. Esta dupla es el corazón de la solución: instruye al navegador (y al motor de renderizado PDF) para que preserve los espacios y sangrías del código original, pero forzando un salto de línea de forma agresiva si la cadena de texto amenaza con romper el margen físico del contenedor.

Como capa de seguridad adicional (*Graceful Degradation*), se añadió `overflow-x: auto;` y `max-width: 100%;` de forma exclusiva a la etiqueta `<pre>`. Así, si por alguna limitación estructural el salto de línea fallase, el bloque generaría su propia barra de desplazamiento horizontal independiente, protegiendo la estructura principal del documento móvil.

El Aprendizaje / Deuda Técnica

El Diseño Web Responsivo (*Responsive Web Design*) a menudo olvida las etiquetas técnicas. Este hallazgo refuerza la metodología de "Especificación como Fuente" (*Spec as Source*): los estilos CSS no solo deben embellecer el HTML, sino protegerlo bajo cualquier contexto de lectura.

Proveer reglas CSS agnósticas que cubran tanto las ventanas de los navegadores móviles como el renderizado de impresión física (PDFs estáticos) consolida la dualidad de nuestra documentación. Hemos erradicado un *bug* silencioso y recurrente sin añadir lógica pesada en JavaScript, manteniendo un rendimiento puro de 100/100.