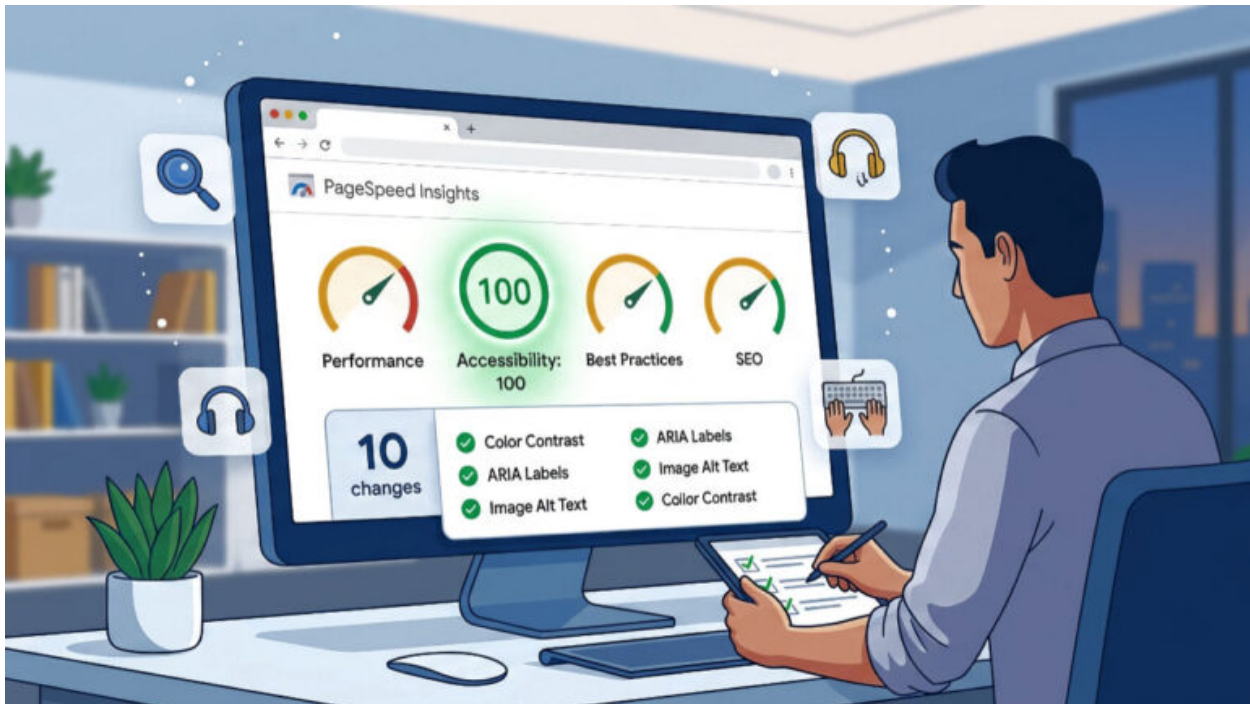


# Andy 21

## ¿Cómo mejorar la accesibilidad de tu web?



¿Sabías que el 80% de los problemas de accesibilidad web se resuelven con menos de diez cambios concretos? Aquí están, con ejemplos reales.

Recientemente se auditaron a fondo varias webs. No porque fueran lentas ni porque tuvieran problemas visibles, sino por rigor técnico. La herramienta utilizada fue **PageSpeed Insights**, que Google ofrece de forma gratuita en [pagespeed.web.dev](https://pagespeed.web.dev).

Lo que se encontró sorprendió. No por el rendimiento, que ya era razonablemente bueno, sino por la accesibilidad. Webs que visualmente funcionaban perfectamente tenían problemas reales que afectaban a usuarios con discapacidades. Y lo mejor: tenían solución concreta. El proceso de identificar y corregir cada problema fue además más rápido y sistemático gracias a la ayuda de la **inteligencia artificial**.

# Qué mide PageSpeed Insights

PageSpeed Insights analiza cualquier URL y devuelve una puntuación de 0 a 100 en cuatro categorías, tanto para escritorio como para móvil:

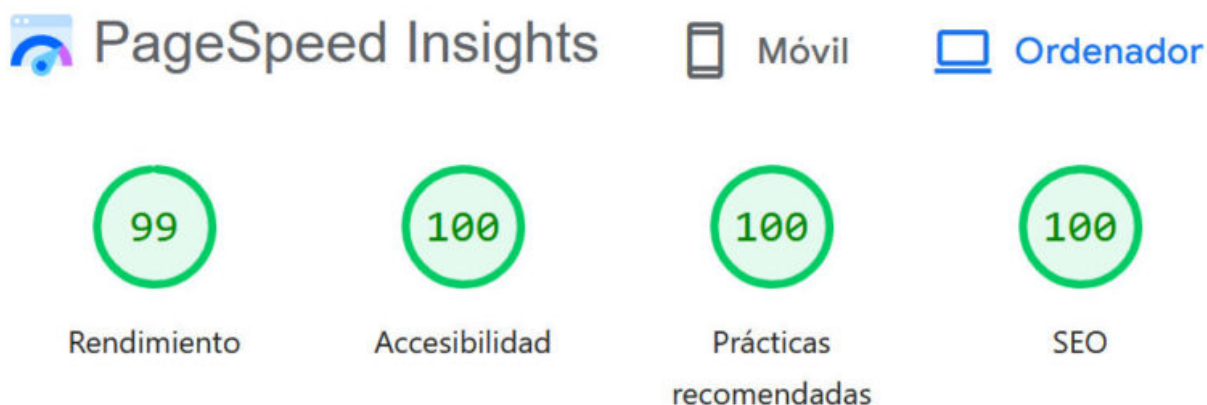
**Rendimiento** – velocidad de carga real, medida con métricas como LCP (tiempo hasta que se renderiza el elemento más grande visible), FCP (primer contenido visible), CLS (estabilidad visual mientras carga) e INP (respuesta a la interacción). Es la categoría más compleja de optimizar y la más dependiente de factores externos.

**Accesibilidad** – si la web es usable por personas con discapacidades visuales, motoras o cognitivas. Contraste de colores, etiquetas en formularios, textos alternativos en imágenes, estructura de encabezados, atributos ARIA. Es donde más se puede mejorar con cambios concretos y controlados.

**Buenas prácticas** – seguridad y estándares modernos: HTTPS, ausencia de librerías con vulnerabilidades conocidas, ausencia de errores en consola, uso correcto de APIs del navegador.

**SEO técnico** – si la web es rastreable e indexable: meta descriptions, títulos únicos, enlaces con texto descriptivo, viewport configurado, robots.txt y muchos otros factores onpage.

De las cuatro, la accesibilidad es la que más se entrelaza con el resto y la que más impacto tiene en el conjunto cuando se trabaja bien.



# El estándar detrás: WCAG y la ley

Google no se ha inventado nada. Los criterios que mide PageSpeed están basados en las **WCAG** (Web Content Accessibility Guidelines), las pautas internacionales publicadas por el W3C. La referencia actual es WCAG 2.1 nivel AA.

En España esto no es opcional para muchas webs. El **Real Decreto 1112/2018** transpone la Directiva Europea 2016/2102 y obliga a todas las webs del sector público a cumplir WCAG 2.1 nivel AA desde 2020. Las webs privadas que prestan servicios públicos o reciben financiación pública también están afectadas. Y aunque las webs comerciales privadas no tienen obligación legal directa por ahora, la tendencia regulatoria europea apunta claramente hacia una extensión progresiva.

*Cumplir WCAG AA significa que tu web funciona para el 15-20% de la población que tiene algún tipo de discapacidad. No es un tecnicismo menor.*

## Los problemas más frecuentes que encontré

### Contraste de colores insuficiente

WCAG AA exige una relación de contraste mínima de 4,5:1 entre texto y fondo para texto normal, y 3:1 para texto grande. Es el problema más frecuente y el más fácil de pasar por alto porque visualmente puede parecer perfectamente legible.

En la práctica: un **#0080FF** sobre fondo blanco tiene un ratio de aproximadamente 3,1:1, insuficiente. Cambiándolo a **#0059B3** el ratio sube a 4,6:1 y supera el mínimo. El cambio visual es casi imperceptible para la mayoría, pero para alguien con baja visión o daltonismo puede ser la diferencia entre leer o no leer el contenido. **WebAIM Contrast Checker** permite verificar cualquier combinación antes de aplicarla.

### Botones sin etiqueta accesible

Un botón que solo contiene un icono sin texto visible necesita un atributo **aria-label** para que los lectores de pantalla puedan

describirlo. Sin él es completamente opaco para un usuario con discapacidad visual.

```
<button type="button" aria-label="Copiar al portapapeles">  
  <svg>...</svg>  
</button>
```

## Imágenes sin atributo alt

El atributo **alt** es obligatorio según WCAG y es uno de los factores de SEO on-page más básicos. Una imagen sin **alt** es invisible para Google y para cualquier lector de pantalla. El texto debe describir el contenido o función de la imagen, no ser un relleno genérico.

## Enlaces sin texto descriptivo

Un enlace que dice «más información» sin contexto no aporta nada ni al usuario con lector de pantalla ni a Google. La solución es usar texto descriptivo en el atributo **title** y añadir **aria-label** cuando el texto visible no puede cambiarse:

```
<a href="/servicio-seo" title="Posicionamiento Web SEO" aria-label="Más  
información sobre el servicio de Posicionamiento Web SEO">  
  Más información  
</a>
```

## Estructura de encabezados e iframes

Los encabezados H1, H2, H3 no son solo estilo visual, son estructura semántica. Un H3 antes de un H2, o varios H1 en la misma página, confunden tanto a los lectores de pantalla como a Google. La jerarquía debe ser lógica y coherente. En la misma línea, un iframe sin atributo **title** es opaco para la accesibilidad: tanto Google Maps como cualquier contenido embebido necesita un título descriptivo.

```
<iframe title="Mapa de localización de nuestra oficina"  
src="..."></iframe>
```

# Accesibilidad, SEO y UX: las tres patas del trípode

Trabajando en estas mejoras se hace evidente que accesibilidad, SEO

on-page y experiencia de usuario no son disciplinas separadas. Son la misma cosa vista desde ángulos distintos. Al igual que ocurre con la patas de un trípode, si una falla las otras se desequilibran.

*Cada mejora de accesibilidad repercute positivamente en el SEO técnico y en la experiencia general. No es trabajo doble, es trabajo que rinde en tres frentes a la vez.*

## **Webs muy distintas, mismo resultado**

Las webs que se auditaron y optimizaron son técnicamente muy distintas: una WordPress con múltiples plugins, otra en PHP puro sin ningún CMS, una con publicidad de AdSense, otra sin ella. Perfiles completamente diferentes.

*El resultado en accesibilidad fue el mismo en todas: 100 sobre 100. Lo que cambia de un proyecto a otro es el camino, no el destino.*

En WordPress algunos problemas se resuelven con configuración, otros requieren código personalizado. En PHP puro el control es total. Con publicidad aparecen elementos de terceros que complican el análisis. Pero los principios son siempre los mismos: los que dicta WCAG AA.

## **Lo que no siempre se puede controlar**

El rendimiento móvil es la categoría más difícil de optimizar y la más dependiente de factores externos. Las webs administradas tienen entre 73 y 90 sobre 100 en rendimiento móvil según el momento de la medición, variabilidad normal porque PageSpeed combina datos reales de usuarios con mediciones de laboratorio.

Hay casos donde el techo es real: scripts de redes publicitarias como AdSense o iframes de servicios externos introducen código de terceros sobre el que no hay control y que penaliza el rendimiento móvil. Se puede optimizar todo lo propio y aun así no llegar al 100 sin eliminar funcionalidad valiosa. En esos casos la decisión correcta es documentar el límite y explicar por qué existe.

*La accesibilidad en cambio no tiene esa excusa. Depende casi en su totalidad de decisiones propias de desarrollo y diseño, y se puede llegar al 100 en cualquier web si se trabaja con rigor.*

# Cómo empezar y cómo puede ayudarte Andy

Si quieres auditar tu web: entra en [pagespeed.web.dev](https://pagespeed.web.dev), introduce tu URL y revisa la sección de Accesibilidad. Los problemas aparecen listados con el elemento concreto que falla. Por donde empezar: contraste de colores, botones con solo icono, **aria-label** donde corresponda, imágenes sin **alt** y enlaces sin texto descriptivo en **title**. Son los cinco aspectos que con más frecuencia se pueden mejorar y también los que más puntuación suman cuando se corrigen.

Y si prefieres delegar, Andy puede auditar tu web, identificar exactamente qué falla y aplicar las correcciones. Es el mismo trabajo que describe este post, con el mismo objetivo: llegar al 100 en accesibilidad independientemente de cómo esté construida tu web.

---

Si te ha gustado este PDF, compártelo. Si no te ha gustado, díselo a [Andy](#).