

Optimizaciones de Rendimiento para PageSpeed 100

Optimizaciones Implementadas

1. CSS Crítico Minimizado

- CSS inline reducido a esenciales en `layout.tsx`
- Solo incluye reset mínimo y placeholder LCP

2. Headers de Caché Óptimos (`vercel.json`)

- **Imágenes:** 1 año (`max-age=31536000, immutable`)
- **Estáticos (_next/static):** 1 año
- **Fuentes:** 1 año
- **Favicons/Manifest:** 1 día con revalidación

3. Browserslist Moderno

Eliminado IE11 para reducir polyfills:

```
"browserslist": [
  "Chrome >= 90",
  "Firefox >= 90",
  "Safari >= 14",
  "Edge >= 90",
  "iOS >= 14",
  "not IE 11"
]
```

4. Imágenes Responsivas

- Componente `ResponsiveImage` con `srcset` automático
- Versiones generadas: 400w, 800w, 1200w, 1600w
- Hero optimizado con `preload` `srcset`

5. Dynamic Imports

- Componentes below-the-fold con `next/dynamic`
- Reducción del bundle inicial

Recomendaciones Adicionales (Cloudflare/CDN)

Habilitar Compresión Brotli

En Cloudflare Dashboard:

1. Speed → Optimization → Content Optimization
2. Activar “Brotli” compression

Configurar Caché en Cloudflare

Page Rules para imágenes:

URL: gruasequier.com/images/*
Cache Level: Cache Everything
Edge Cache TTL: 1 month
Browser Cache TTL: 1 year

Minimizar Recursos de Terceros

1. Considerar cargar beacon.js de Cloudflare solo después de interacción
2. Lazy-load scripts no críticos

Métricas Objetivo

- **LCP:** < 2.5s (actualmente ~8s → reducir imagen hero)
- **FCP:** < 1.8s (objetivo < 1s)
- **TBT:** < 200ms (ya óptimo ~30ms)
- **CLS:** 0 (ya óptimo)
- **Speed Index:** < 3.4s

Próximos Pasos

1. Comprimir más las imágenes del hero (quality 75-80%)
2. Evaluar AMP para páginas de blog
3. Implementar Service Worker para caché offline