

INSTRUCCIÓN CRÍTICA: Necesito que apliques las optimizaciones de vite.config.ts que reportaste como "Cannot Apply". Esos cambios son ESENCIALES para llegar de 87 a 100 en PageSpeed.

ACLARACIÓN: vite.config.ts NO es un archivo protegido en este proyecto. Es un archivo de configuración estándar que DEBE ser modificado para optimizar el build de producción.

PASO 1: Instalar Critters

Ejecuta este comando exactamente como está escrito:

```
```bash
```

```
npm install --save-dev critters
```

PASO 2: Modificar vite.config.ts REEMPLAZA el contenido completo de vite.config.ts con esta configuración optimizada:

```
import { defineConfig } from 'vite';
import react from '@vitejs/plugin-react';
import path from 'path';
export default defineConfig({
 plugins: [react()],
 resolve: {
 alias: {
 '@': path.resolve(__dirname, './src'),
 },
 },
 build: {
 target: 'es2015',
 cssCodeSplit: true,
 minify: 'terser',
 terserOptions: {
 compress: {
 drop_console: true,
 drop_debugger: true,
 pure_funcs: ['console.log', 'console.info'],
 passes: 2,
 },
 mangle: {
 safari10: true,
 },
 format: {
 comments: false,
 },
 },
 rollupOptions: {
 output: {
 manualChunks(id) {
 // Separar React y React-DOM en su propio chunk
 if (id.includes('node_modules/react/') ||
id.includes('node_modules/react-dom/')) {
 return 'vendor-react';
 }

 // Separar TanStack Query
 if (id.includes('@tanstack/react-query')) {
```



```

 return 'vendor-query';
 }

 // Separar lucide-react icons
 if (id.includes('lucide-react')) {
 return 'vendor-icons';
 }

 // Resto de node modules
 if (id.includes('node_modules')) {
 return 'vendor-misc';
 }
},
chunkFileNames: 'assets/[name]-[hash].js',
assetFileNames: 'assets/[name]-[hash].[ext]',
},
},
chunkSizeWarningLimit: 600,
reportCompressedSize: true,
},
server: {
 port: 5173,
 strictPort: false,
 proxy: {
 '/api': {
 target: 'http://localhost:3000',
 changeOrigin: true,
 },
 },
},
});

```

PASO 3: Actualizar index.html con Critical CSS Inline Agrega este bloque INMEDIATAMENTE DESPUÉS del y ANTES de cualquier :

```

<style>
/* Critical CSS - Above the fold */
*,::before,::after{box-sizing:border-box;border-width:0;border-style:solid;border-color:#e5e7eb}
html{line-height:1.5;-webkit-text-size-adjust:100%;font-family:Inter,ui-sans-serif,system-ui,sans-serif}
body{margin:0;line-height:inherit}
h1,h2,h3,h4,h5,h6{font-size:inherit;font-weight:inherit}
a{color:inherit;text-decoration:inherit}
img,svg,video{display:block;max-width:100%}
button,input,select,textarea{font-family:inherit;font-size:100%;line-height:inherit;color:inherit;margin:0;padding:0}
button{background-color:transparent;background-image:none;cursor:pointer}
.dark{color-scheme:dark}
.dark body{background-color:#111827;color:#f9fafb}
</style>

```

PASO 4: Modificar el de CSS principal CAMBIA esta línea:



```
<link rel="stylesheet" href="/src/index.css" />
```

POR esta (con media="print" para diferir la carga):

```
<link rel="stylesheet" href="/src/index.css" media="print"
onload="this.media='all'" />
```

```
<noscript><link rel="stylesheet" href="/src/index.css" /></noscript>
```

PASO 5: Rebuild con optimizaciones activas

```
npm run build
```

VALIDACIÓN POST-BUILD: Después del build, verifica en la terminal que veas outputs como:

✓ dist/assets/vendor-react-[hash].js (40-50 KiB) ✓ dist/assets/vendor-query-[hash].js (10-15 KiB) ✓ dist/assets/vendor-icons-[hash].js (15-20 KiB) ✓ dist/assets/index-[hash].js (50-60 KiB)

<- Este debe ser MÁS PEQUEÑO que antes

PASO 6: Preview local antes del deploy

```
npm run preview
```

Abre <http://localhost:4173> y verifica:

1. La página carga correctamente
2. No hay errores de consola
3. El hero section aparece inmediatamente
4. Las secciones inferiores aparecen al hacer scroll

PASO 7: Deploy a producción Una vez verificado localmente, haz el deploy.

PASO 8: Re-test PageSpeed Después del deploy (espera 2-3 minutos para que el CDN se actualice):

1. Ve a: <https://pagespeed.web.dev/>
2. Ingresa: <https://grupotranservica.com>
3. Selecciona "Mobile"
4. Ejecuta el análisis

RESULTADO ESPERADO:

- Puntuación: 95-100 (antes 87)
- FCP: 1.0-1.5s (antes 2.5s)
- LCP: 1.5-2.0s (antes 3.5s)
- JavaScript inicial: ~60 KiB (antes 121 KiB)

DEBUGGING (si algo falla): ✗ Error "Cannot find module 'path'" → Ejecuta: npm install --save-dev @types/node ✗ Build falla con "manualChunks" → Verifica que hayas copiado la configuración exacta ✗ Página en blanco después del deploy → Revisa la consola del navegador, probablemente sea un path incorrecto ✗ CSS no carga → Verifica que el snippet de diferir CSS esté correcto (con media="print")