IA EN EL CLIENT-SIDE

El costo oculto en tu INP (y cómo evitarlo)



Web Workers · WebGPU · Chunking · Streaming

QUIÉN SOY

Leslie Herrera

Software Engineer · Web & AI · Speaker · Community Builder

De la banca y las finanzas a la ingeniería de software. Hoy construyo en la intersección de Frontend + IA, buscando interfaces claras, accesibles y útiles.



Finanzas → Tech

5 años en banca antes de escribir código.

LATAM Airlines

Software Engineer · Next.js + Node.js

Comunidad

Women Techmakers Ambassador · AWS Community Builder · JSConf Chile

DATOS FREAK

- 🩰Ballet
- 🥋Taekwondo
- 🎤Aprendiendo a cantar
- 🎮Overwatch
- 🐕Perros

@dereemii · Santiago de Chile

¿A quién le han pedido usar más IA en el trabajo simplemente porque sí?

EL ROL ESTÁ CAMBIANDO

¿Qué cambia en el trabajo frontend?

La IA acelera el código repetitivo. El criterio de producto sigue en manos humanas.

IA GENERA

- Componentes y layouts
- Código base
- Variantes simples

LAS PERSONAS DECIDEN

- Qué necesita el usuario
- Accesibilidad y performance
- Arquitectura de sistemas



El reto: sumar IA sin romper la experiencia.

POR QUÉ NOS IMPORTA EL PERFORMANCE

10–20s

Es lo que tarda un usuario en decidir si se queda o se va.

Con una propuesta de valor clara en esos primeros 10 segundos, esa ventana se puede extender a varios minutos de atención real.

Cada interacción lenta es una cuenta regresiva en tu contra — por eso hablamos de INP.

CASO REAL

redBus redujo su INP en **72%** (menos scroll handlers, menos actualizaciones de estado) y vio un **+7%** en ventas.

LA SIGUIENTE PREGUNTA

Si la IA ya existe en la nube, ¿para qué correrla en el navegador?



Privacidad

Tus datos se quedan **en el dispositivo.**



Costo

Menos llamadas a servicios de inferencia.



Latencia

Respuesta local, sin round-trip al servidor.



Offline

Funciona sin conexión, una vez cargado el modelo.

HERRAMIENTAS | Transformers.js · ONNX Runtime Web · WebLLM

Ya es real: Chrome prueba Gemini Nano; también hay demos de transcripción, eliminación de fondos y chat local.

El reto no es ejecutar IA local: es integrarla sin romper la experiencia.

@dereemii

CORE WEB VITALS

¿Qué es el INP?

Es una de las métricas de Core Web Vitals —junto a LCP y CLS— y mide cuánto pasa desde que haces clic, tap o presionas una tecla hasta que el navegador pinta la siguiente actualización visual, durante toda la visita. Te califica casi por tu peor interacción, no por el promedio.



Presionar una tecla

Física o en pantalla.



Tocar la pantalla

Un tap dispara pointerdown, pointerup y click — cuenta como una sola interacción.



Hacer clic

Con un mouse, en cualquier elemento.

EN LA PRÁCTICA

Así se siente (y se mide) un mal INP

Escribes "zapatillas" en un buscador con autocompletado:

BUENO

Las sugerencias aparecen letra por letra, casi al mismo tiempo que escribes. Se siente instantáneo.



MALO

El cursor se adelanta a las letras. Dudas si tu tecla funcionó y vuelves a presionarla — ahora hay texto duplicado.



BUENO	NECESITA MEJORA	MALO
≤ 200ms	200–500ms	> 500ms

Un modelo corriendo en el main thread empuja tu app directo a la zona roja — durante toda la visita, no solo al cargar.

CONCEPTO CLAVE

¿Qué es el main thread?

El hilo que ejecuta el JavaScript de tu página, calcula estilos, arma el layout, pinta la pantalla y procesa los eventos de la interfaz.

- **Ejecutar JavaScript**
- **Calcular estilos (CSS)**
- **Layout (reflow)**
- **Pintar y componer**
- **Responder a eventos del usuario**

50 ms



Es lo que basta para convertir cualquier tarea en un **"long task"** — y un forward pass de un modelo lo logra fácil, si no lo sacas del hilo principal.

ANATOMÍA DEL INP

Las tres fases de una interacción

1. INPUT DELAY

El hilo está ocupado

El navegador no puede ni empezar a procesar el evento porque el main thread sigue corriendo otra cosa.



2. PROCESSING TIME

Ejecutar el handler

Aquí es donde vive un forward pass de inferencia si lo llamas directo desde el click handler.



3. PRESENTATION DELAY

Pintar el resultado

Streaming ataca justo esta fase: mostrar algo antes de tener todo listo.



dereemii

DEMO 01

> lesliieherrera.dev/inp

INP EN TIEMPO REAL

El mismo trabajo, tres formas de ejecutarlo.

— Medido con Event Timing API



01 LAS HERRAMIENTAS · 1 DE 4

Chunking & cooperative scheduling

Parte el trabajo en pedazos pequeños y cede el turno al navegador entre cada uno, en vez de ejecutar todo de corrido.

- El cambio más barato de implementar
- No requiere mover código a otro hilo
- Reduce, pero no elimina, el bloqueo

```
// ceder el turno entre chunks
for (const chunk of tokens) {
  process(chunk);
  await scheduler.yield();
}
```

soporte parcial — fallback: setTimeout(fn, 0)



Para tareas menores en el hilo principal. Si necesitas aislar la inferencia por completo, el siguiente paso es Web Workers →

Web Workers

Un hilo de JavaScript aparte del principal, sin acceso al DOM — ahí puede correr la inferencia sin bloquear la UI. Pero no basta con moverla: cómo le pasas los datos del modelo también importa.

postMessage normal

Structured clone

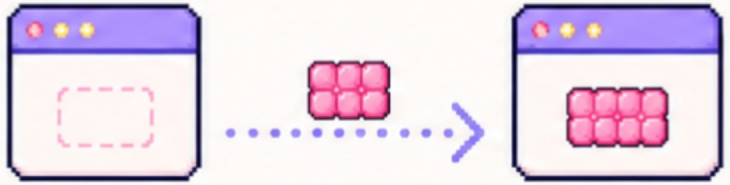


queda duplicado en ambos lados

Copia el buffer completo entre hilos. Con pesos de un modelo, eso es costoso.

Transferable Objects

Zero-copy



se mueve — ya no existe en el origen

Un ArrayBuffer se mueve, no se copia. postMessage(buf, [buf]).

Para modelos grandes, transferir buffers evita copias costosas.

WebGPU

La API que le da al navegador acceso directo a la GPU, pensada para compute y no solo para gráficos. No saca tu JavaScript del hilo principal por sí sola — eso lo hace el Worker; WebGPU acelera el cómputo compatible una vez que ya estás ahí.

FALLBACK

WASM



Corre en CPU, portable, buen soporte.

FALLBACK

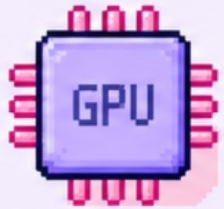
WebGL



GPU vía shaders gráficos, no pensado para ML.

IDEAL

WebGPU



Compute shaders diseñados para esto.

Soporte todavía parcial en 2026 — siempre necesitas una ruta de fallback.

04 LAS HERRAMIENTAS · 4 DE 4

Streaming

Muestra resultados token por token, sin esperar la respuesta completa. Mejora la velocidad percibida; el impacto en INP depende del trabajo de cada actualización.

SIN STREAMING

La pantalla parece congelada hasta que termina la generación.



CON STREAMING

TTFT · tiempo hasta el primer token

El primer token aparece rápido. El trabajo total puede tardar lo mismo.



ReadableStream + backpressure = resultados incrementales.

PRO TIP • Evita medir layout (por ejemplo, offsetHeight) en cada token: puede causar layout thrashing y borrar la ganancia del streaming.

ARQUITECTURA DE REFERENCIA

Tres capas, un solo trabajo cada una

MAIN THREAD

UI

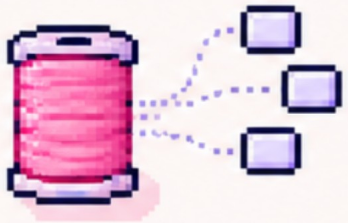


Solo renderiza y responde a eventos. Nunca ejecuta inferencia.

transferable

WORKER

Orquestación

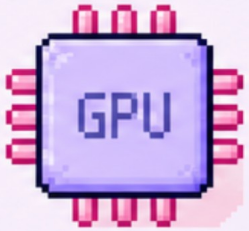


Tokeniza, hace chunking, coordina el streaming de resultados.

compute

GPU

WebGPU



Ejecuta el forward pass del modelo vía compute shaders.

Cada fase de la interacción vive en una capa distinta — y ninguna bloquea la que pinta tu UI.

dereemii

DEMO 02

> lesliieherrera.dev/llm

LLM EN EL NAVEGADOR



distilgpt2 generando texto, token por token.

— Hugging Face · Transformers.js

MAIN THREAD

WORKER + WASM

WORKER + WEBGPU

SEAMOS HONESTOS

El costo oculto de la solución



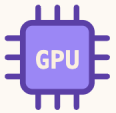
Arranque de un Worker

Tiene su propio overhead: no lo uses para trabajo trivial.



Cache del modelo

Pesos de MB o GB: necesitas Cache API / OPFS para no re-descargar.



Soporte de WebGPU

Todavía parcial entre navegadores: siempre necesitas fallback.



Debugging

Errores cruzando hilos son más difíciles de rastrear.

Vale la pena, pero mídelo: ninguna solución es gratis.

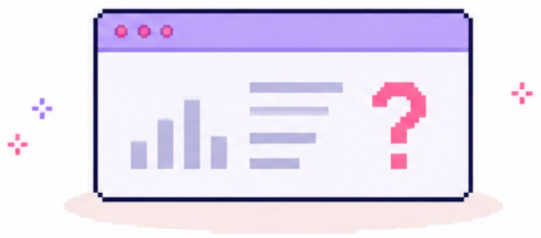
AHORA, EN TU PROPIA PÁGINA

Cuando PageSpeed no tiene el dato

PAGESPEED · CrUX

No hay datos suficientes

Faltan interacciones reales para calcular un INP representativo.



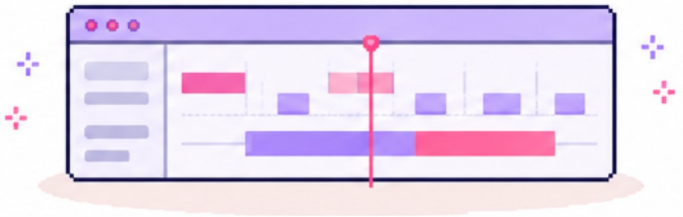
DIAGNÓSTICO LOCAL · DEVTOOLS

Reproduce la interacción

Vista móvil · CPU 4× · panel Performance

Interactions → input, procesamiento y presentación

Main Thread → la tarea larga que bloquea



Para INP cuentan clicks, taps y teclas; scroll y mouse-move no.

Así validas en tu página lo que acabas de ver en vivo.

TAKEAWAYS

dereemii

01 La IA puede afectar cada interacción, no solo la carga.

02 El INP refleja cómo se siente tu producto.

03 Chunking, Workers y streaming ayudan a liberar el main thread.

04 El criterio conecta herramientas, producto, performance y UX.

RECETA · Worker + transferables + streaming + mide tu INP



Más IA no garantiza mejores productos.
Mejores decisiones crean mejores experiencias.

¡Gracias!

leslieherrera.dev

> **leslieherrera.dev/inp**
Demo 01 · INP en tiempo real

> **leslieherrera.dev/llm**
Demo 02 · LLM en el navegador



escanea · demos y slides

Leslie Herrera · Santiago, Chile