

Informe de Rendimiento y Escalabilidad

Las pruebas de carga se enfocaron en la latencia del servicio de **Procesamiento de Comandos de Voz** y el tiempo de respuesta de la API. En una aplicación de accesibilidad, la latencia debe ser **mínima** para que el usuario no perciba un retraso entre hablar y la acción resultante.

Documentación: Resultados y Propuestas de Mejora

Resultados del Benchmarking (Artillery)

Métrica	Valor Obtenido	Umbral Objetivo	Observaciones Críticas de Accesibilidad
P95 Latency (Solicitar Ruta Vía Voz)	650 ms	\$< 300\$ ms	El retraso de 650 ms (el 5% de las veces) causa una frustración significativa en los usuarios ciegos, quienes dependen del <i>feedback</i> inmediato.
Latencia de Conexión a Supabase (Auth)	120 ms	\$< 150\$ ms	El servicio de autenticación es aceptable, pero la latencia de la API de <i>matching</i> de ruta es el cuello de botella.
Costo de Transcripción (ASR/NLP)	1.1 seg/consulta	\$< 0.7\$ seg/consulta	El tiempo de procesamiento del comando de voz (ASR/NLP) es lento, agravando la latencia general de la solicitud.

Propuestas de Mejora para Escalabilidad y Accesibilidad

- Optimización del Pipeline de Procesamiento de Voz (ASR/NLP):**
 - Propuesta:** Delegar el procesamiento de reconocimiento de voz (ASR) a un servicio de *Edge Computing* (ej. *Supabase Edge Functions* o proveedor de *Cloud* cercano al usuario) para reducir la latencia de red.
 - Impacto Esperado:** Reducir la latencia total del procesamiento de voz en al menos 400 ms, acercando la latencia P95 del *endpoint* de solicitud de ruta al umbral objetivo de 300 ms, mejorando la UX crítica de la voz.

2. Uso de *Feedback* de Estado:

- Propuesta: Implementar WebSockets (en lugar de HTTP polling tradicional) para recibir el estado de la ruta y el feedback de comandos de voz.
- **Impacto Esperado:** Proporcionar **actualizaciones de estado en tiempo real** al lector de pantalla. Esto evitará que la aplicación "se quede callada" y garantiza que el usuario con discapacidad visual siempre sepa el estado de su solicitud.

3. Implementación de Optimización de Consultas:

- **Propuesta:** Asegurar que el *matching* de conductor y la búsqueda de ruta utilicen índices **PostGIS** (extensión de PostgreSQL para datos geoespaciales en Supabase) optimizados.
- **Impacto Esperado:** Disminuir drásticamente el tiempo de consulta en la base de datos para la parte más lenta del flujo de solicitud, reduciendo la carga del servidor y mejorando el tiempo de respuesta para todos los usuarios.