



Universidad Autónoma de Nayarit



Unidad académica de economía

Carrera:

Sistemas computacionales

Materia:

Programación del lado del cliente

Tema:

Reto IA 6

Maestro:

Eligardo Cruz Sánchez

Alumna:

Erika Alejandra Orozco Vázquez

Reto IA 6

| Criterio | REST + Polling (cada 5s) | WebSocket | Server-Sent Events (SSE) | Mi evaluación |
|--|---|--|---|--|
| Complejidad en el cliente (1–5) | 1 – fetch + setInterval, cero dramas | 4 – manejo de conexión, eventos, reconexión | 2 – EventSource, casi plug & play | Polling es el mas simple y sencillo, tanto WebSockets como SSE suele ser un poco mas llamativo para el cliente |
| Uso de recursos del servidor (1000 clientes) | Alto $1000 \times (12 \text{ req/min}) = 12\,000 \text{ req/min}$ aunque no cambie nada | Medio 1000 conexiones abiertas, pero solo envía cuando hay cambios | Bajo-Medio 1000 conexiones HTTP persistentes, solo push | Polling es de trabajo constante; WebSocket y SSE solo trabajan cuando hay cambios. |
| Latencia del cambio de inventario | Hasta 5 s (depende del intervalo) | Muy baja casi instantánea | Muy baja push inmediato | polling pierde por que suele ser el más lento. WebSocket y SSE suelen ser muy rápidas |
| Pérdida breve de conexión | Tolerante la siguiente petición “lo arregla” | Más delicado hay que reintentar y resincronizar estado | Muy buena reconexión automática + Last-Event-ID | SSE reconecta solo y continúa el stream; WebSocket y Polling requieren intervención |
| Facilidad con mock server | Muy fácil JSON Server, JSONPlaceholder, Express | Media necesitas soporte WS real | Fácil Express / NestJS lo soportan rápido | WebSocket requiere infraestructura real; SSE y polling se realizan fácil con json y express |

Mi elección: Para algo personal y de pruebas yo escogería Polling , ya que es el más fácil y sencillo de hacer , pero para algo más profesional escogería el SSE ya es fácil de crear y mantener y además de que más llamativo a la vista del cliente.