

Análisis técnico:

1. Desglose del rendimiento servicio por servicio

Escenario	Samples	P50(ms)	P95(ms)	P99(ms)
1. User Login	100	90	105	109
3. Browse Products	100	65	104	107
4. Get Product Details	100	30	56	60
Create Cart	100	39	57	59
Add Item	100	107	147	154
Get Items	100	33	57	58
Update Item	100	74	105	107
Delete Item	100	34	53	58
Create Order	100	314	397	406
Get Order Status	100	27	54	59
Create Payment	100	523	922	953
Get Payment	100	7	49	53

2. Identificación de cuellos de botella mediante métodos sistemáticos

Por tiempo de respuesta:

. Create Order y Create Payment. Tienen un P95 y P99 Alto. Esto lleva a que el proceso de negocio sea lento, puede que esté bloqueando recursos.

Por porcentaje de Error: Muy alto en User Login, Create Cart, Create Order.

Escenario	Error %
1. User Login	0,00%
3. Browse Products	0,00%
4. Get Product Details	0,00%
Create Cart	0,00%
Add Item	0,00%
Get Items	0,00%
Update Item	0,00%
Delete Item	0,00%

Create Order	16,00%
Get Order Status	16,00%
Create Payment	50,00%
Get Payment	66,00%

3. Cálculos de planificación de la capacidad con factores de seguridad

Usuarios máximos = $0.10114 \times 5000 = 505.7$

Usuarios máximos alcanzables: 506

Capacidad recomendada (70 %): 354 usuarios concurrentes

Conclusión: bajo un factor de seguridad del 70 %, el sistema puede manejar de forma segura aproximadamente 354 usuarios concurrentes, muy por debajo de los 5 000 requeridos, por lo que no cumple con el objetivo de capacidad.

4. Validación del cumplimiento de SLA.

.User Login

P95 = 105 ms. Cumple objetivo (≤ 200 ms).

P99 = 109 ms. Cumple objetivo (≤ 500 ms).

Tasa de error = 0.00 %. Cumple objetivo ($< 0.1\%$).

Throughput = 0.101 req/s. No cumple objetivo 1000 RPS.

.Browse Products

P95 = 104 ms. Cumple objetivo (≤ 300 ms).

P99 = 107 ms. Cumple objetivo (≤ 500 ms).

Tasa de error = 0.00 %. Cumple objetivo.

Throughput = 0.10101 req/s. No cumple objetivo 2000 RPS.

.Get Product Details

P95 = 56 ms. Cumple objetivo (≤ 150 ms).

P99 = 60 ms. Cumple objetivo (≤ 400 ms).

Tasa de error = 0.00 %. Cumple objetivo.

Throughput = 0.10101 req/s. No cumple objetivo 1200 RPS.

.Create Cart

P95 = 57 ms. Cumple objetivo (≤ 200 ms).

P99 = 59 ms. Cumple objetivo (≤ 400 ms).

Tasa de error = 0.00 %. Cumple objetivo ($< 0.2\%$).

Throughput = 0.10101 req/s. No cumple objetivo 1500 RPS.

.Add Item

P95 = 147 ms. Cumple objetivo (≤ 200 ms).

P99 = 154 ms. Cumple objetivo (≤ 400 ms).

Tasa de error = 0.00 %. Cumple objetivo ($< 0.2\%$).

Throughput = 0.101 req/s. No cumple objetivo 1500 RPS.

.Get Items

P95 = 57 ms. Cumple objetivo (≤ 150 ms).

P99 = 58 ms. Cumple objetivo (≤ 400 ms).

Tasa de error = 0.00 %. Cumple objetivo.

Throughput = 0.10101 req/s. No cumple objetivo 1200 RPS.

.Update Item

P95 = 105 ms. Cumple objetivo (≤ 150 ms).

P99 = 107 ms. Cumple objetivo (≤ 400 ms).

Tasa de error = 0.00 %. Cumple objetivo.

Throughput = 0.10101 req/s. No cumple objetivo 1200 RPS.

.Delete Item

P95 = 53 ms. Cumple objetivo (≤ 150 ms).

P99 = 58 ms. Cumple objetivo (≤ 400 ms).

Tasa de error = 0.00 %. Cumple objetivo.

Throughput = 0.10102 req/s. No cumple objetivo 1200 RPS.

.Create Order

P95 = 397 ms. Cumple objetivo (≤ 500 ms).

P99 = 406 ms. Cumple objetivo (≤ 800 ms).

Tasa de error = 16.00 %. No cumple objetivo < 0.1 %.

Throughput = 0.10102 req/s. No cumple objetivo 500 RPS.

.Get Order Status

P95 = 54 ms. Cumple objetivo (≤ 150 ms).

P99 = 59 ms. Cumple objetivo (≤ 400 ms).

Tasa de error = 16.00 %. No cumple objetivo < 0.1 %.

Throughput = 0.10105 req/s. No cumple objetivo 500 RPS.

.Create Payment

P95 = 922 ms. Cumple objetivo (≤ 1000 ms).

P99 = 953 ms. Cumple objetivo (≤ 2000 ms).

Tasa de error = 50.00 %. No cumple objetivo < 0.05 %.

Throughput = 0.10106 req/s. No cumple objetivo 200 RPS.

.Get Payment

P95 = 49 ms. Cumple objetivo (≤ 150 ms).

P99 = 53 ms. Cumple objetivo (≤ 400 ms).

Tasa de error = 66.00 %. No cumple objetivo < 0.1 %.

Throughput = 0.10114 req/s. No cumple objetivo 800 RPS

5. Análisis de correlación entre servicios

Existe una cadena de dependencia clara en los endpoints Create Order, Get Order Status, Create Payment. Como estos endpoints utilizan datos en común y tienen una dependencia directa entre el dato generado por el servicio anterior, al fallar la generación del caso el servicio siguiente falla.

6. Análisis de patrones de error e investigación de causas fundamentales

Patrones detectados:

El error que más se repite es el error 429/Too Many Requests, lo cual representa que la api tiene un límite en cuanto a la cantidad de llamadas que soporta por minuto.

7. Recomendaciones sobre el dimensionamiento de la infraestructura

Primero que nada se debe realizar modificaciones en el backend de la api, ampliando la cantidad de llamadas permitidas por minuto.

Por otro lado, los servicios de órdenes y catálogo presentan los mayores tiempos de respuesta bajo carga. Se recomienda revisar consultas a base de datos y aplicar índices