

ARTI 4109 - Arquitecturas de Software

Cuaderno de Trabajo

Proyecto

Grupo 5

Camilo Alejandro Nossa Calderón
Brayan Felipe Rojas Bernal
Elmar Santofimio Suarez
Juan David Forero Rodríguez
Carlos Fidel Rodríguez Alarcón



Reto – Cliente Grupo 2

Reto: Una empresa de transporte necesita modernizar su plataforma digital para soportar la gestión de reservas, el monitoreo en tiempo real de su flota y la comunicación entre conductores y pasajeros. Actualmente, la empresa opera 2,500 vehículos en 8 ciudades, manejando alrededor de 150,000 reservas diarias y picos de máximo 10,000 usuarios concurrentes. El proyecto debe garantizar alta disponibilidad, pues la arquitectura actual mantiene una disponibilidad del 97.5% que es significativamente por debajo del objetivo del 99.95% requerido para competir efectivamente en el mercado, donde cada minuto de inactividad representa pérdidas aproximadas de \$8,000 USD y deterioro de la confianza del cliente. A su vez, el crecimiento acelerado del negocio exige escalabilidad, incrementando la complejidad técnica para soportar aumentos inesperados de demanda expresados en los 10,000 usuarios mencionados anteriormente. La seguridad de datos personales es un aspecto crítico pues la plataforma maneja información sensible de más de 3 millones de usuarios registrados, incluyendo datos de tarjetas de crédito tokenizadas, historiales de viaje, y ubicaciones en tiempo real que deben cumplir con regulaciones. También, es necesario asegurar una integración confiable con aplicaciones móviles, sistemas internos, pasarelas de pago y aliados externos, considerando riesgos de compatibilidad y sincronización. Por último, la facilidad de modificación es esencial para responder a regulaciones y condiciones operativas que cambian entre ciudades, como variaciones en tarifas, restricciones de circulación y requerimientos de accesibilidad.

Cliente: Empresa de transporte G2

- Gestión de reservas.
- Monitoreo en tiempo real de su flota.
- Comunicación entre conductores y pasajeros.



Seguridad	
Unidad:	Segundos
Respuesta esperada:	< 0.3 segundos

Actor:	Actor externo a la organización	Estímulo:	Registro de una reserva
Ambiente:	Operación normal	Artefacto:	Servicio de reservas
Respuesta esperada:	Detectar un cambio no autorizado en la reserva en menos de 0.3 segundos.		



Latencia	
Prioridad	Media
Impacto	Baja



Disponibilidad	
Unidad:	Milisegundos
Respuesta esperada:	$\leq 800\text{ms}$

Actor:	Sistema monitoreo	Estímulo:	Monitorear los componentes de registro de reservas.
Ambiente:	Operación normal	Artefacto:	Sistema de reservas
Respuesta esperada:	Detectar posibles fallos en los componentes como en la información enviada del registro de reservas en un tiempo igual o inferior a 800 milisegundos.		



Latencia	
Prioridad	Media
Impacto	Baja

Latencia	
Unidad:	Segundos
Respuesta esperada:	≤ 3 segundos

Escalabilidad	
Unidad:	(# de reservas/minuto) * 60 minuto
Respuesta esperada:	10.000/minuto * 60 minutos

Actor:	Usuario	Estímulo:	Hacer una reserva
Ambiente:	Operación normal (1000 reservas por minuto)	Artefacto:	Sistema de reservas
Respuesta esperada:	Crear una reserva en un tiempo igual o inferior a 3 segundos.		

Seguridad	
Prioridad	Baja
Impacto	Baja

Disponibilidad	
Prioridad	Baja
Impacto	Baja



Latencia	
Unidad:	Segundos
Respuesta esperada:	≤ 1 segundo

Actor:	Usuario	Estímulo:	Procesar el pago de una reserva
Ambiente:	Operación normal	Artefacto:	Módulo de pagos
Respuesta esperada:	Procesar el pago en un tiempo igual o inferior a 1 segundo.		



Escalabilidad	
Prioridad	Media
Impacto	Media



Disponibilidad	
Unidad:	Segundos
Respuesta esperada:	< 1 segundo

Actor:	Sistema de monitoreo	Estímulo:	Recibir ubicación de un vehículo (flota)
Ambiente:	Operación normal	Artefacto:	Servicio monitoreo
Respuesta esperada:	Detectar cuando el proceso no actualice la ubicación de una flota de manera correcta en un tiempo inferior a 1 segundo.		



Latencia	
Prioridad	Baja
Impacto	Baja



Seguridad	
Unidad:	Segundos
Respuesta esperada:	< 0.5 segundos

Actor:	Usuario no autorizado	Estímulo:	Modificar información de una ruta
Ambiente:	Operación normal	Artefacto:	Servicio de monitoreo
Respuesta esperada:	Detectar cualquier intento de modificación de la información de rutas realizado por un usuario no autorizado, en menos de 0.5 segundos.		



Latencia	
Prioridad	Baja
Impacto	Baja



Facilidad de integración	
Unidad:	Horas/ Hombre
Respuesta esperada:	100

Actor:	Área de operaciones	Estímulo:	Integrar un nuevo servicio externo de publicidad hotelera
Ambiente:	Desarrollo	Artefacto:	Componente de marketing
Respuesta esperada:	Se espera integrar la publicidad de un hotel para el cliente en 100 horas/hombre.		



Facilidad de modificación	
Unidad:	Horas/ Hombre
Respuesta esperada:	60

Actor:	Área de operaciones	Estímulo:	Modificar un nuevo servicio externo de publicidad hotelera
Ambiente:	Desarrollo	Artefacto:	Componente de marketing
Respuesta esperada:	Se espera modificar la publicidad de un hotel para el cliente en 60 horas/hombre.		



Latencia	
Unidad:	Milisegundos
Respuesta esperada:	Retornar rutas disponibles en ≤ 500 ms

Escalabilidad	
Unidad:	(# de consultas/minuto)* 60 minuto
Respuesta esperada:	30.000/minuto * 60 minutos

Actor:	Usuario	Estímulo:	Consultar disponibilidad de rutas
Ambiente:	Operación normal (3000 consultas por minuto)	Artefacto:	Servicio de consulta de rutas
Respuesta esperada:	Consultar rutas disponibles en un tiempo igual o inferior a 500 milisegundos.		



Seguridad	
Prioridad	Baja
Impacto	Baja

Disponibilidad	
Prioridad	Baja
Impacto	Baja



Seguridad	
Unidad:	Segundos
Respuesta esperada:	0.5 segundo

Actor:	Cualquier usuario diferente a un cliente (pasajero)	Estímulo:	Detección de suplantación de identidad
Ambiente:	Operación normal	Artefacto:	Servicio de pago
Respuesta esperada:	Revocar el acceso al usuario suplantado		



Latencia	
Prioridad	Media
Impacto	Baja



Latencia	
Unidad:	Segundos
Respuesta esperada:	Conductor asignado en menos de 3 segundos

Actor:	Sistema	Estímulo:	Asignar conductor
Ambiente:	Operación normal	Artefacto:	Motor de asignación
Respuesta esperada:	Asignar conductor en ≤ 3 segundos		



Disponibilidad	
Unidad:	MTTR
Respuesta esperada:	Recuperación automática \leq 0.72 minutos

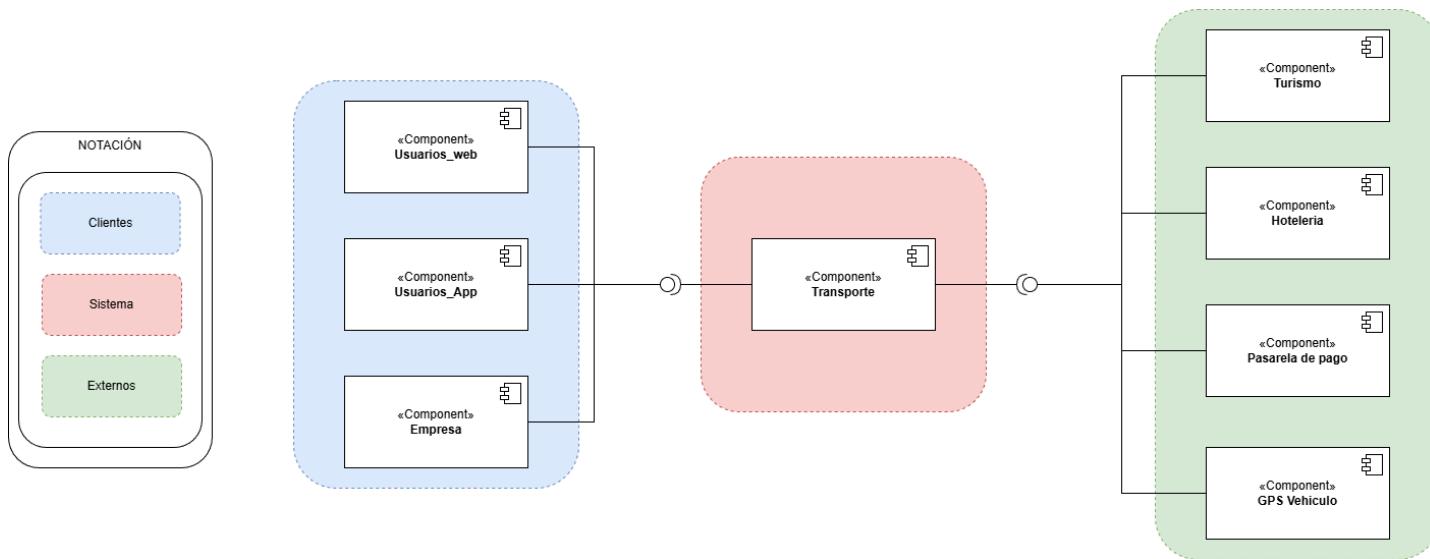
Actor:	Usuario	Estímulo:	Hacer una reserva
Ambiente:	Fallo parcial	Artefacto:	Sistema de reservas
Respuesta esperada:	Recuperación automática \leq 0.72 minutos		

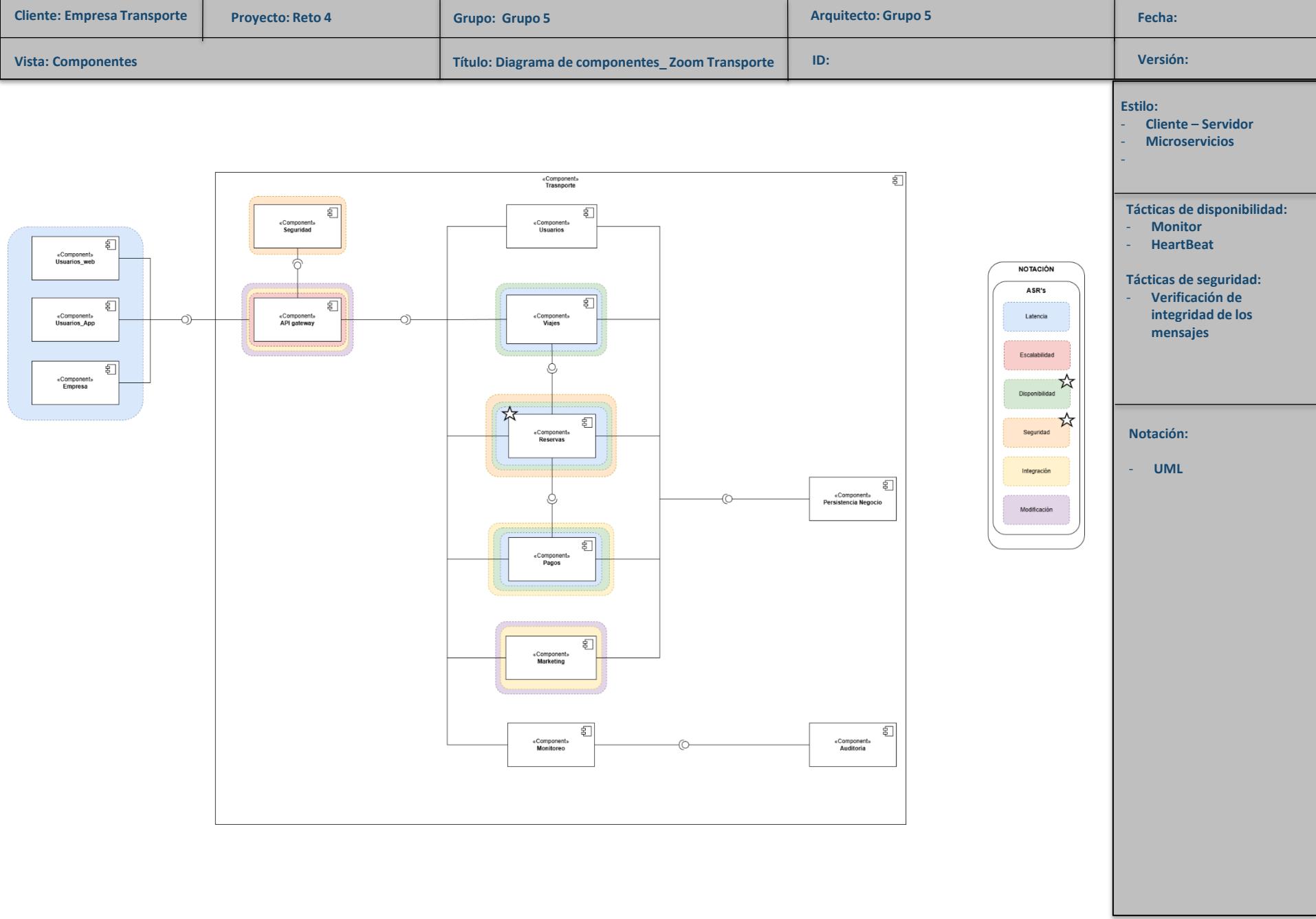


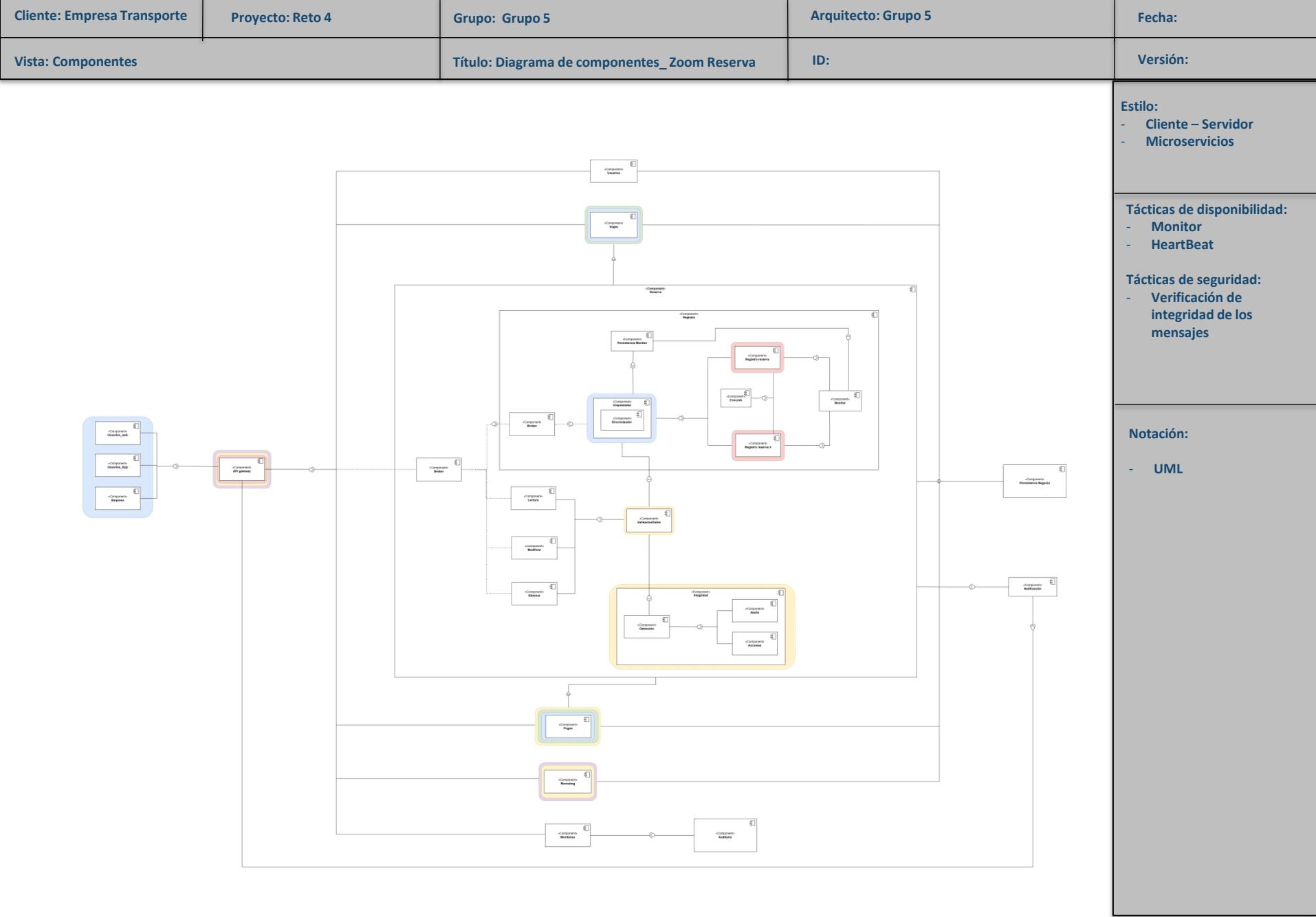
Disponibilidad	
Unidad:	Minutos
Respuesta esperada:	Artefacto en funcionamiento en 60 minutos

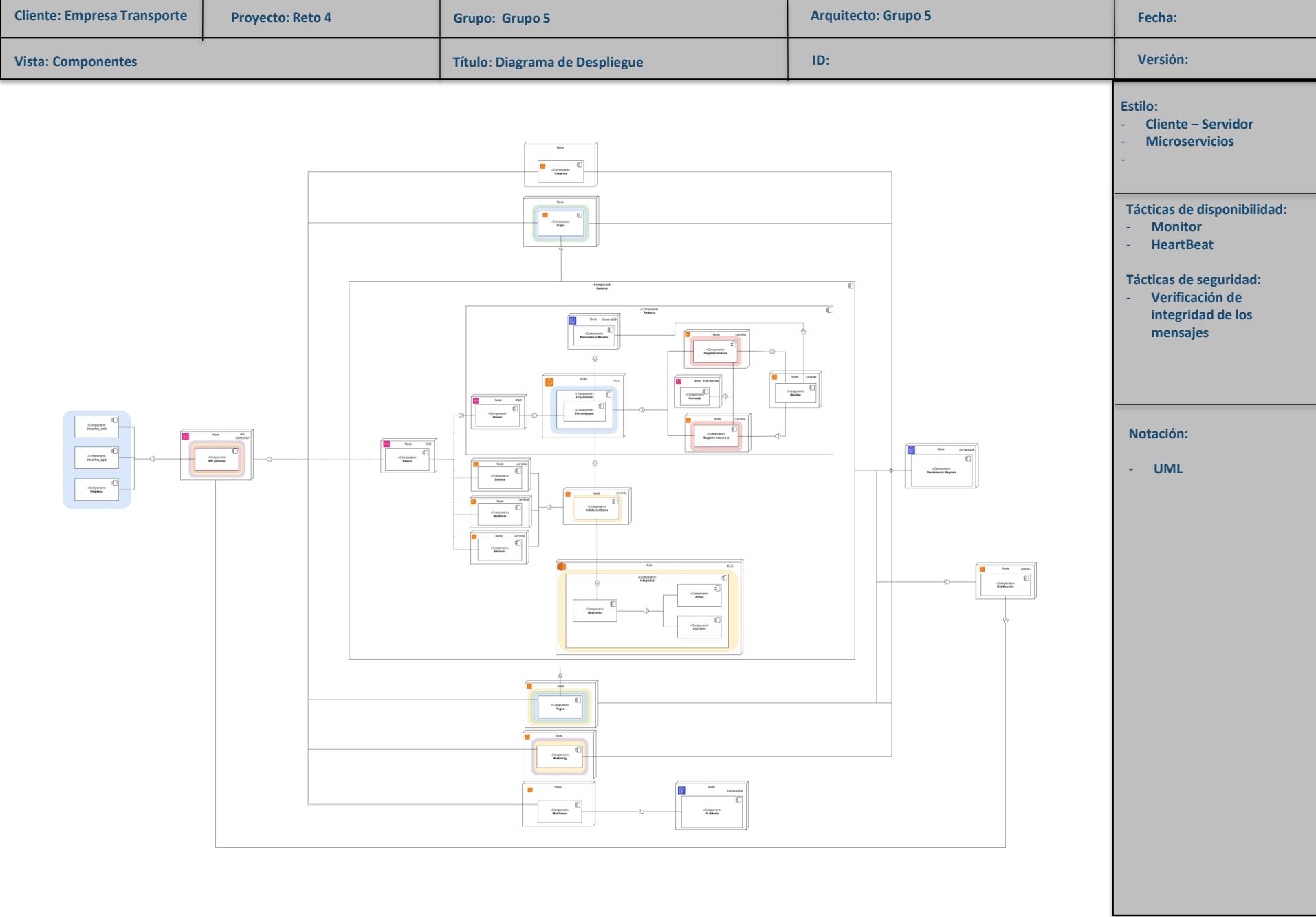
Actor:	Administrador	Estímulo:	Operar durante mantenimiento
Ambiente:	Mantenimiento programado de un artefacto de reservas	Artefacto:	Sistema de reservas
Respuesta esperada:	Reintroducción en 60 minutos		

Cliente: Empresa Transporte	Proyecto: Reto 4	Grupo: Grupo 5	Arquitecto: Grupo 5	Fecha:
Vista: Componentes	Título: Diagrama de contexto	ID:		Versión:
			<p>Estilo:</p> <ul style="list-style-type: none"> - Cliente – Servidor - Microservicios <p>Tácticas de disponibilidad:</p> <ul style="list-style-type: none"> - Monitor - HeartBeat <p>Tácticas de seguridad:</p> <ul style="list-style-type: none"> - Verificación de integridad de los mensajes 	<p>Notación:</p> <ul style="list-style-type: none"> - UML









Experimento

Resultados del experimento

HIPOTESIS:



Detectar amenaza de Tampering en el componente de reserva en menos de 0.3 segundos.



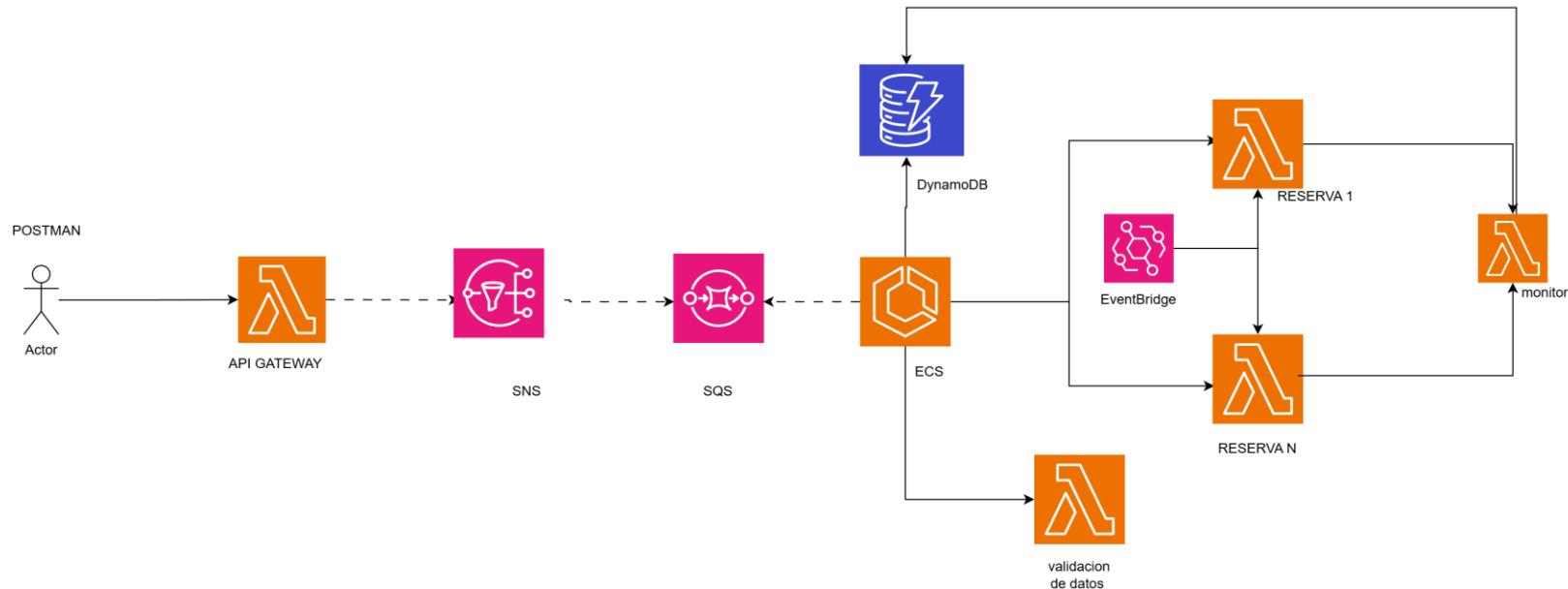
Detectar los nodos disponibles para el registro de reservas en un tiempo igual o inferior a 800 milisegundos.



Crear una reserva en un tiempo igual o inferior a 3 segundos.

Experimento

Arquitectura (AWS)



Experimento

Costos de arquitectura (AWS)



Dynamo DB

Tabla de actualización cada 5 min: muy bajo costo.

Suposiciones:

- 1000/min → 43M writes/mes.
- Writes: $43M \times \$1.25/M = \54 USD .
- Reads: $43M \times \$0.25/M = \11 USD .

Total ≈ \$65 USD.

- 10,000/min → 432M writes/mes.
- Writes: $432M \times \$1.25/M = \540 USD .
- Reads: $432M \times \$0.25/M = \108 USD .

Total ≈ \$650 USD.



Amazon Lambda

Precio:

- \$0.20 por millón de invocaciones + GB-segundos.

Suposiciones:

- 1000/min → 60,000/h → 1.44M/día → ~43M/mes ≈ \$8.6 USD solo por invocaciones.
- 10,000/min → 600,000/h → 14.4M/día → ~432M/mes ≈ \$86 USD solo por invocaciones.
 - El cómputo (memoria/duración) puede añadir entre \$10–50 USD dependiendo de la configuración.

Experimento

Costos de arquitectura (AWS)



ECS

ECS 1 vCPU + 2 GB RAM corriendo 24/7 $\approx \$30\text{--}40$ USD/mes.
Este costo es fijo, no depende de reservas.



EventBridge

2 reglas cada 5 min \rightarrow 576 ejecuciones/día $\rightarrow \sim 17,280$ /mes.
Costo: insignificante (< \$0.02 USD/mes).



SNS

\$0.50 por millón de publicaciones.
1000/min \rightarrow 43M/mes $\rightarrow \sim \$21$ USD.
10,000/min \rightarrow 432M/mes $\rightarrow \sim \$216$ USD.



SQS

Precio:
\$0.40 por millón de solicitudes (Send, Receive, Delete).
Suposiciones:

- 1000/min \rightarrow 60,000/h \rightarrow 1.44M/día $\rightarrow \sim 43$ M/mes.
- $43M \times \$0.40/M = \approx \17 USD/mes.
- 10,000/min \rightarrow 600,000/h \rightarrow 14.4M/día $\rightarrow \sim 432$ M/mes.
- $432M \times \$0.40/M = \approx \173 USD/mes.

Experimento

Costos de arquitectura (AWS)



ECS

ECS 1 vCPU + 2 GB RAM corriendo 24/7 $\approx \$30\text{--}40$ USD/mes.
Este costo es fijo, no depende de reservas.



EventBridge

2 reglas cada 5 min \rightarrow 576 ejecuciones/día $\rightarrow \sim 17,280$ /mes.
Costo: insignificante (< \$0.02 USD/mes).



SNS

\$0.50 por millón de publicaciones.
1000/min \rightarrow 43M/mes $\rightarrow \sim \$21$ USD.
10,000/min \rightarrow 432M/mes $\rightarrow \sim \$216$ USD.



SQS

Precio:
\$0.40 por millón de solicitudes (Send, Receive, Delete).
Suposiciones:

- 1000/min \rightarrow 60,000/h \rightarrow 1.44M/día $\rightarrow \sim 43$ M/mes.
- $43M \times \$0.40/M = \approx \17 USD/mes.
- 10,000/min \rightarrow 600,000/h \rightarrow 14.4M/día $\rightarrow \sim 432$ M/mes.
- $432M \times \$0.40/M = \approx \173 USD/mes.

Experimento

Request del experimento

Request:

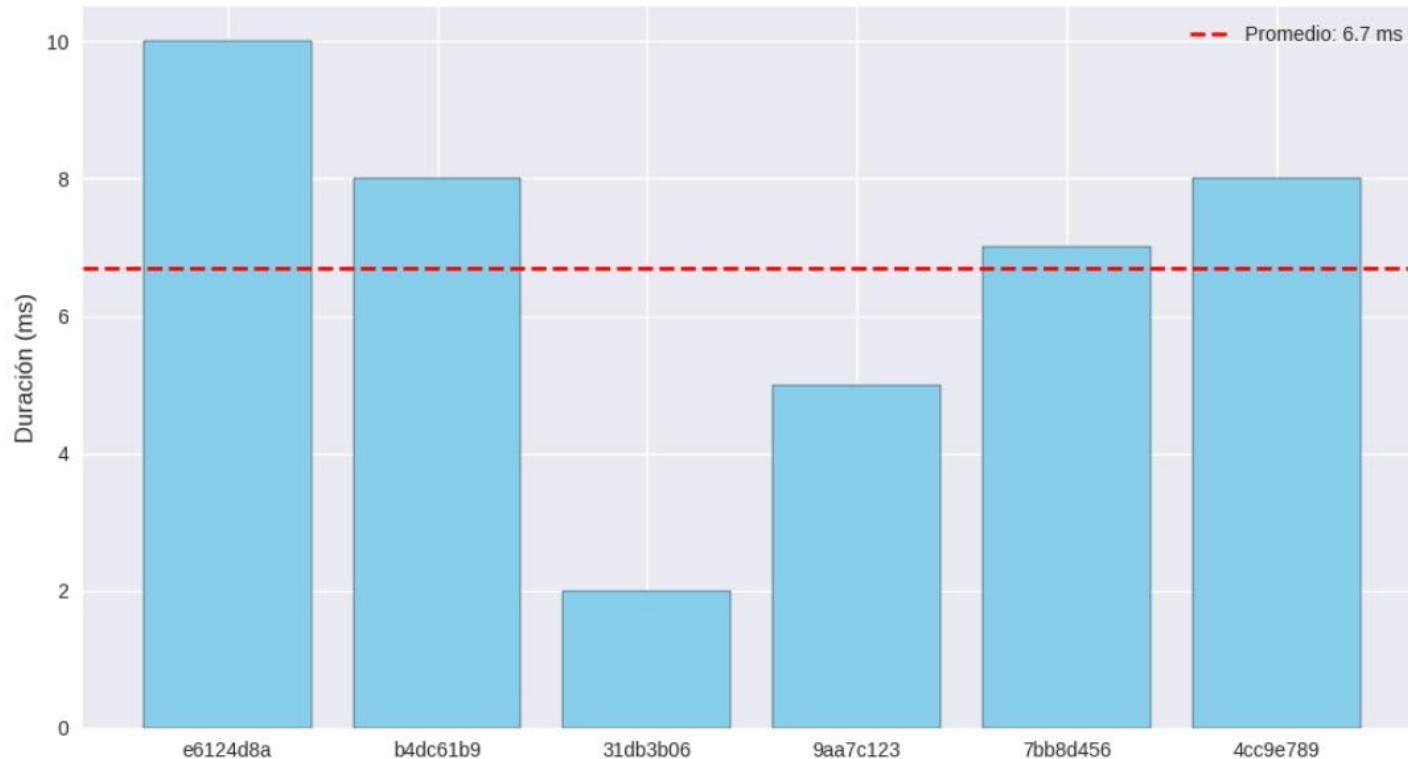
```
{  
    "Asiento": "{$randomInt}"},  
    "BusId": "1324",  
    "CiudadLlegada": "Bogota",  
    "CiudadSalida": "Cajica",  
    "Pago": false,  
    "UsuarioId": "12345"  
}
```



Experimento

Resultados del experimento

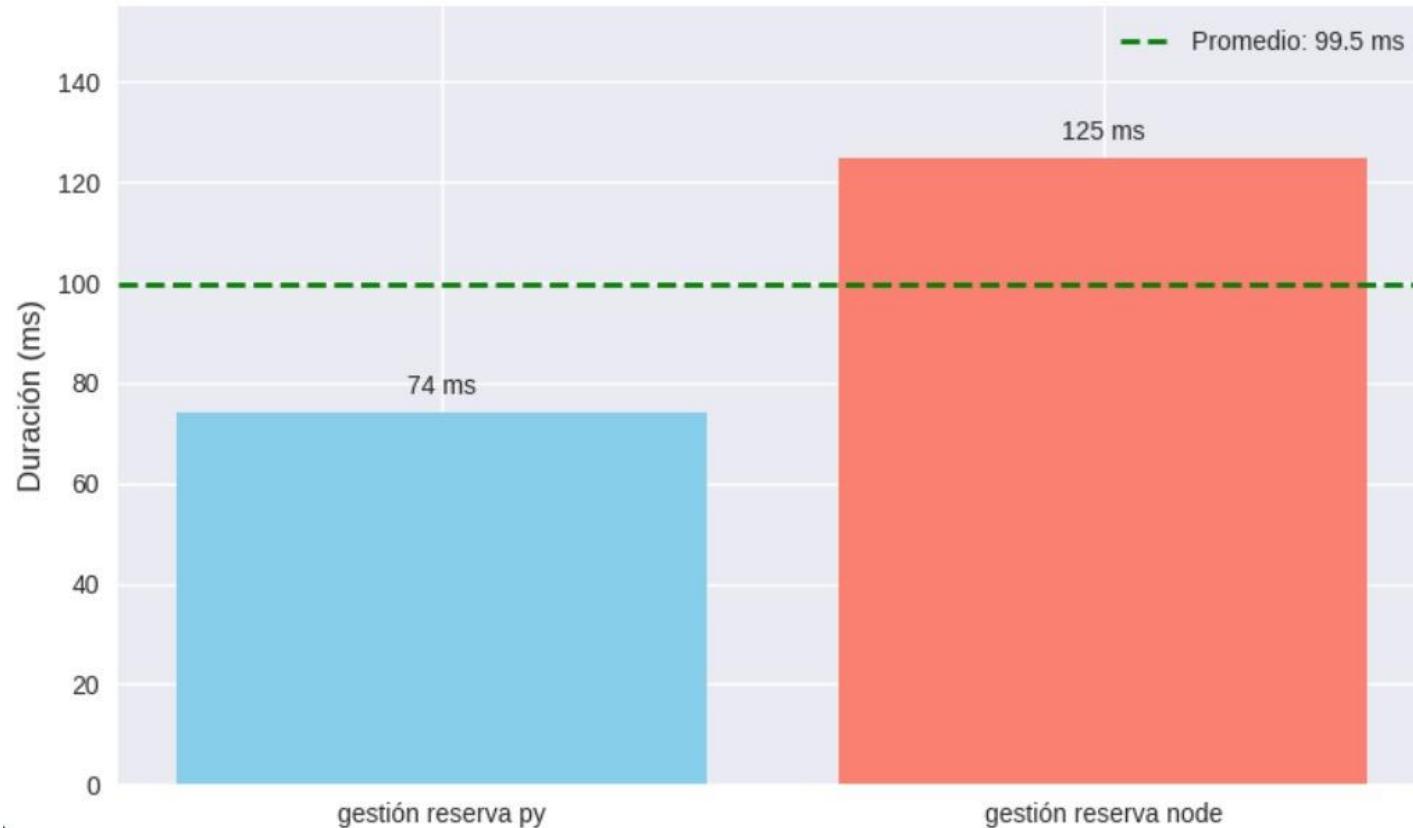
Tiempo validación de datos (seguridad):



Experimento

Resultados del experimento

Tiempo HeartBeat (Disponibilidad) :



Conclusiones

Resultados del experimento

Conclusiones a partir de la experimentación según los ASRs

Seguridad:

La integración de los datos se valida en un tiempo menor al estipulado, lo que demuestra que se cumple con la respuesta esperada. Esto confirma que la infraestructura opera de manera correcta y confiable.

Disponibilidad:

Durante las pruebas en laboratorio observamos que mantener activas las Lambdas de gestión de reservas permite contar con un *heartbeat* rápido. Gracias a ello, se puede verificar con eficacia que la Lambda se encuentra disponible y lista para operar.

Escalabilidad:

La combinación de **Amazon ECS** y las **funciones Lambda** dentro de la infraestructura garantiza que el sistema pueda adaptarse de manera flexible a variaciones en la carga de trabajo. ECS aporta capacidad de procesamiento continuo y estable, mientras que las Lambdas permiten responder de forma dinámica a picos de demanda sin necesidad de aprovisionamiento manual. Esta sinergia asegura que se cumpla con el resultado previsto en el ASR, manteniendo la eficiencia operativa y la capacidad de escalar según las necesidades del negocio.