

MISO

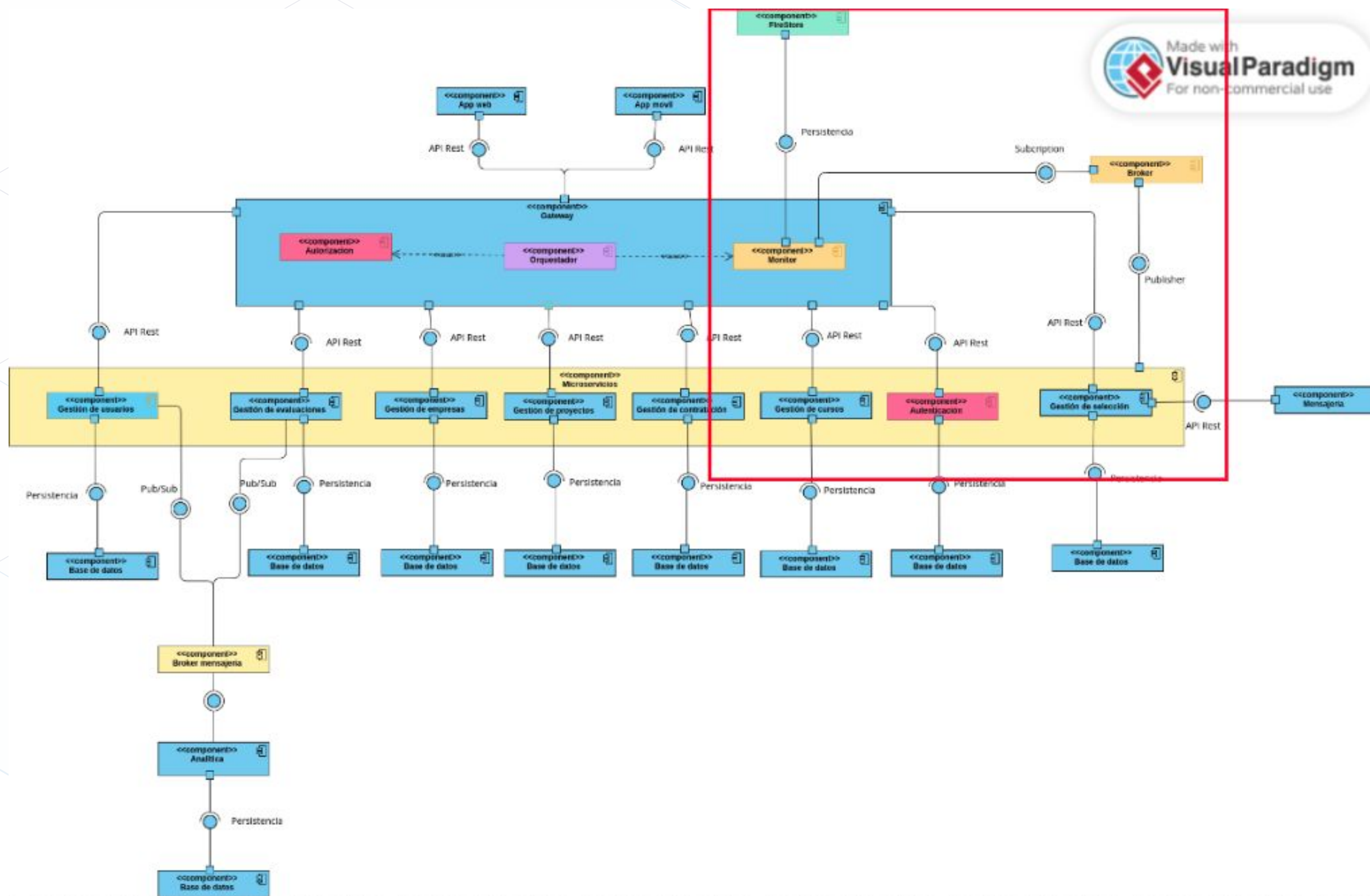
Maestría en Ingeniería de
Software

Experimentación

Título del experimento	HeartBeat
Propósito del experimento	Validar que el componente monitor mediante la evaluación de los últimos estados de salud enviados por el componente de evaluaciones identifica la caída y/o mal funcionamiento del microservicio el 99,9% de las veces.
Resultados Obtenidos	La hipótesis de diseño se cumplió, se logra identificar la caída y/o mal funcionamiento del microservicio el 99,9% de las veces y logra enmascarar el error al usuario.
Esfuerzo total invertido	53 horas/hombre

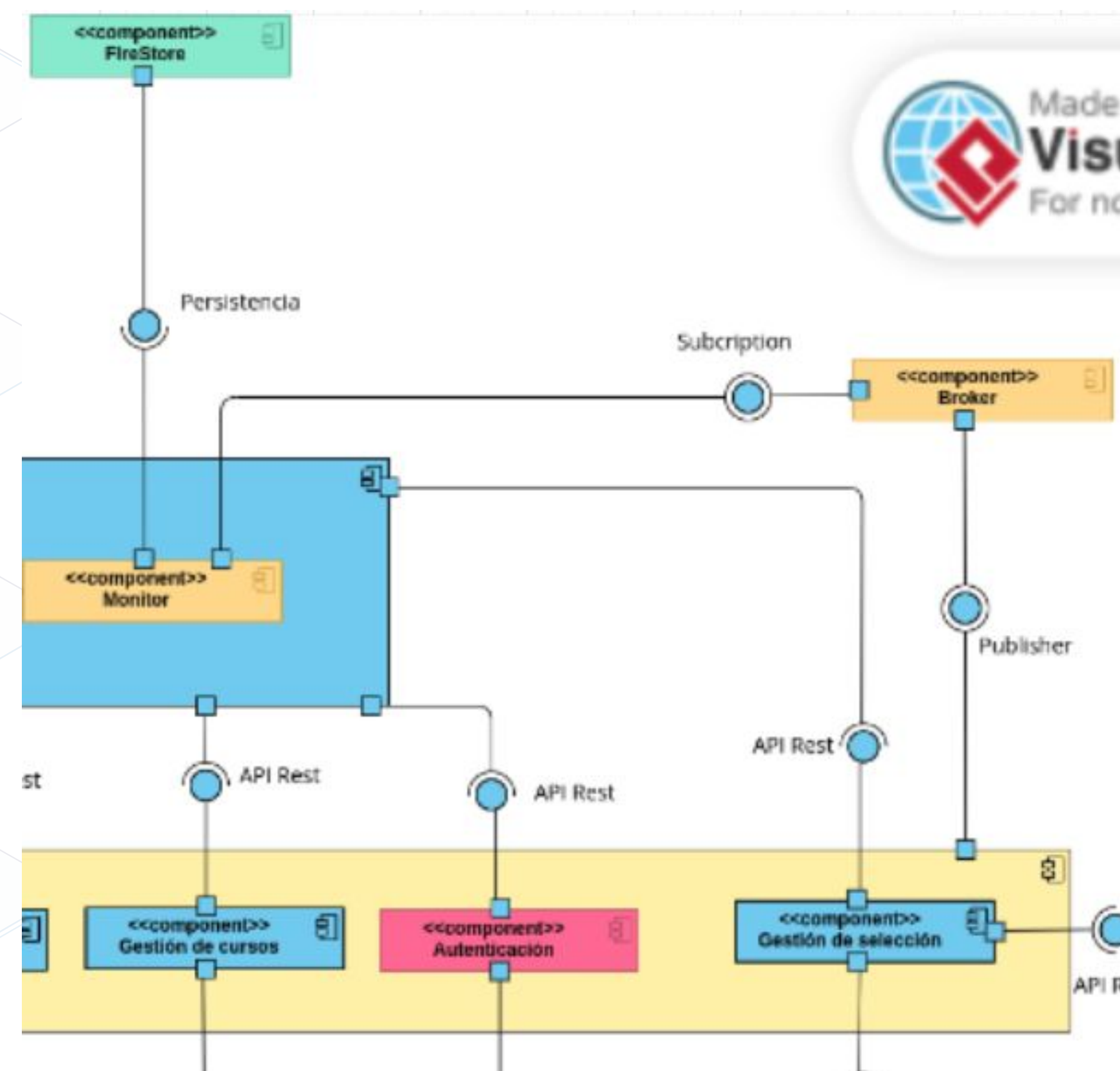


Modelo de componentes



Punto de sensibilidad

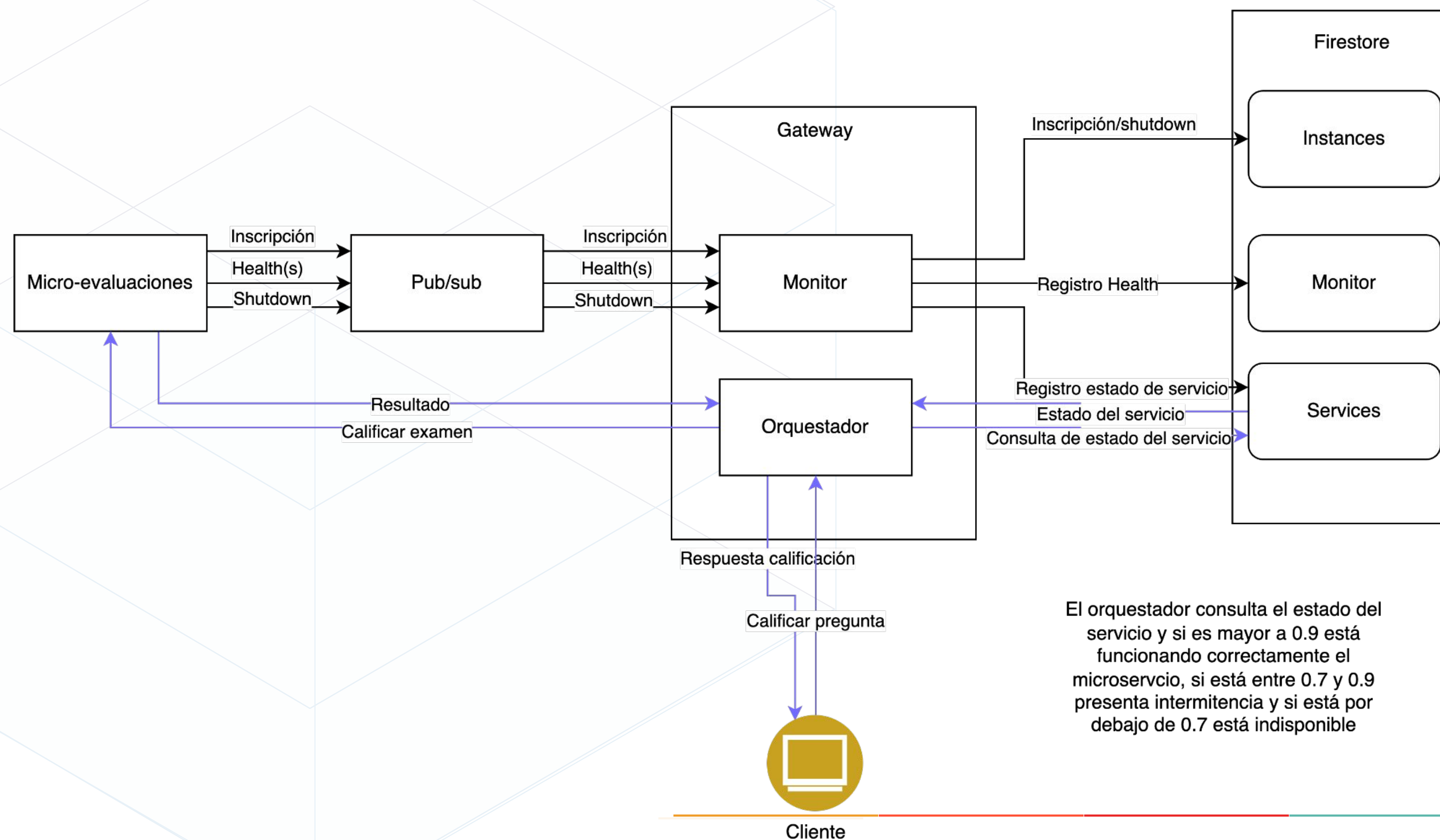
El componente monitor identifica la caída y/o mal funcionamiento del microservicio el 99,9% de las veces para poder enmascarar el error al usuario.

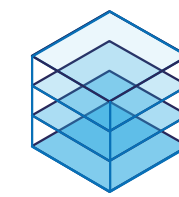


Hipótesis de diseño asociada al experimento	
Punto de sensibilidad	El componente monitor identifica la caída y/o mal funcionamiento del microservicio el 99,9% de las veces para poder enmascarar el error al usuario.
Historia de arquitectura asociada	Como funcionario de ABC, cuando quiera registrar los resultados de una evaluación, dado que el sistema opera con intermitencia, quiero que el sistema identifique la caída y me informe que se presentó un error, esto debe suceder el 99,9% de las veces.
Nivel de incertidumbre	80%, este nivel de incertidumbre de si es posible llegar al 99,9% de caída y/o mal funcionamiento del microservicio.

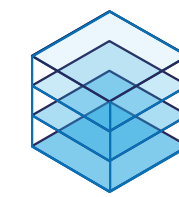


Flujo de información





Ejemplo aplicación en funcionamiento



Tablas de datos

[Datos experimento](#)

Repositorios

Repositorio microservicio de evaluaciones GitHub

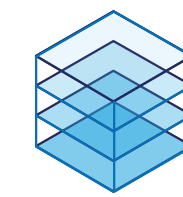
<https://github.com/navi-dupli/microservice-evaluations>

Repositorio gateway GitHub

<https://github.com/navi-dupli/gateway-abc>

Repositorio generador de reporte

<https://github.com/navi-dupli/export-data>



Resultados - Escenario 1

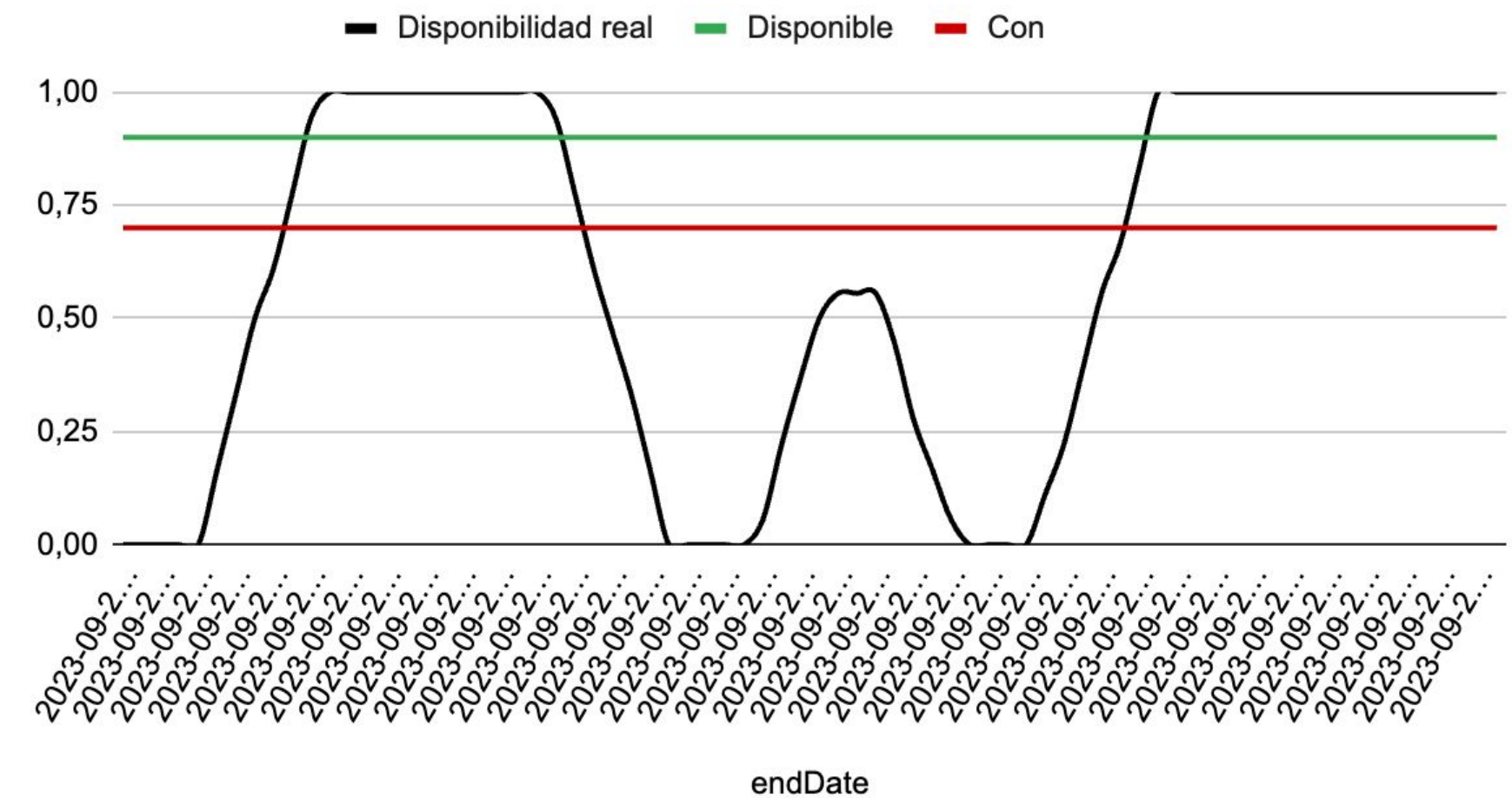
Parametros

Periodicidad envío de señales (s)	Periodicidad evaluación de señales (s)	Instancias activas	Número de peticiones
5	15	1	200

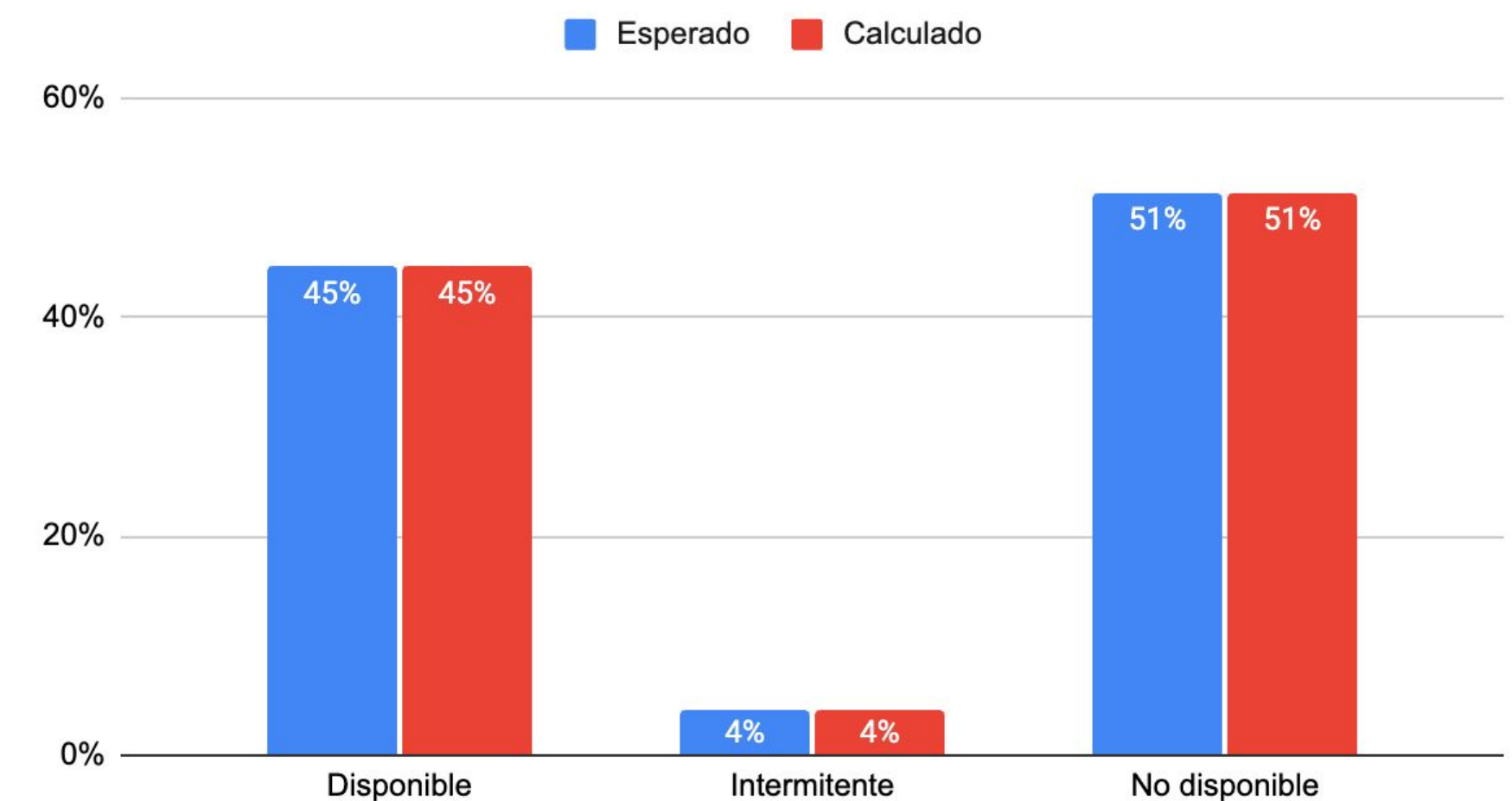
Flujo de la prueba

1. Registro de instancia
2. Envío 50 señales buenas
3. Envío 30 señales malas
4. Envío 10 señales buenas
5. Envío 30 señales malas
6. Envío 50 señales buenas
7. Envío 30 señales malas
8. shutdown de instancia

Indice de disponibilidad del servicio



Esperado Vs Calculado



Resultados - Escenario 2

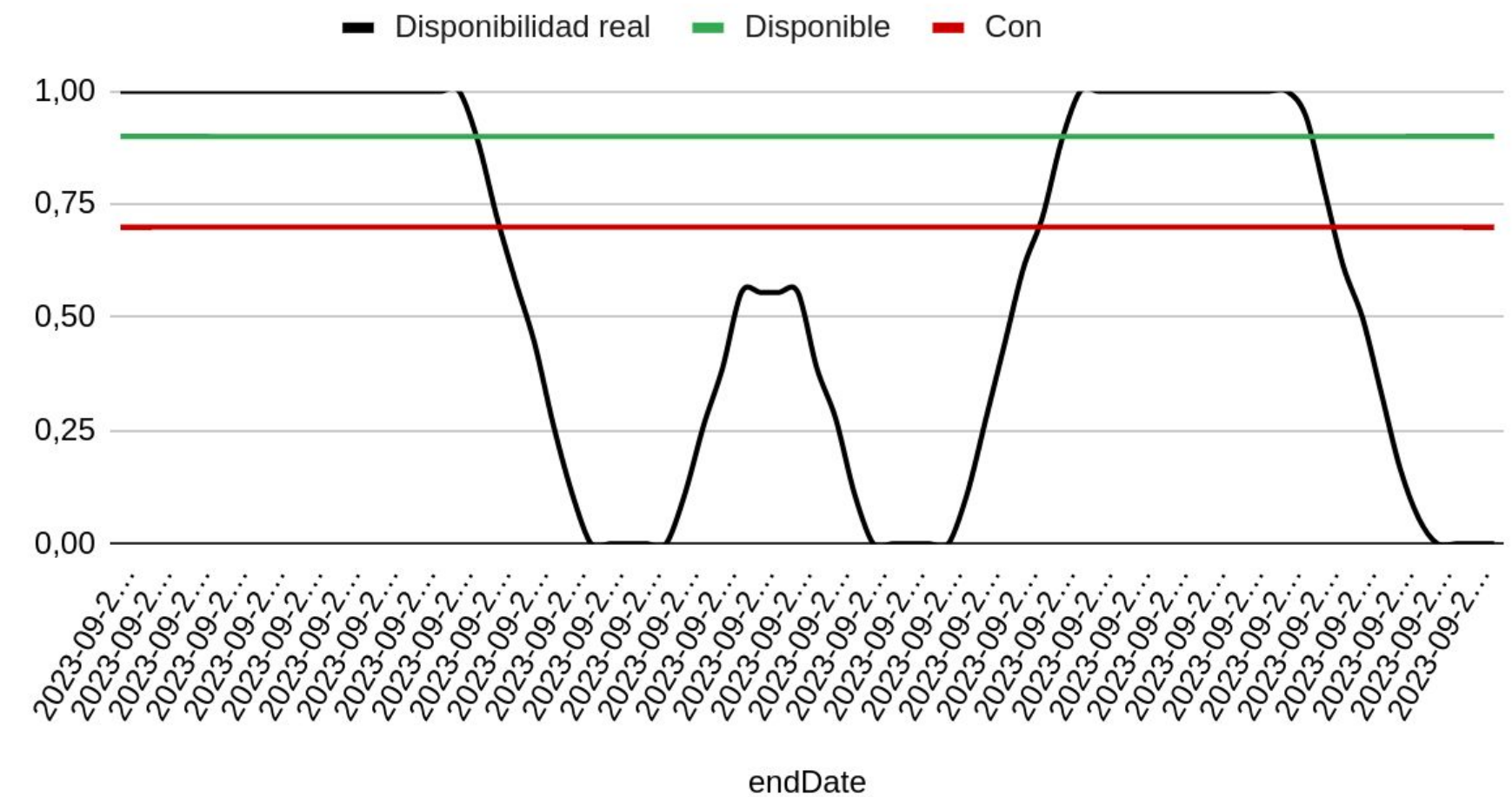
Parametros

Periodicidad envío de señales (s)	Periodicidad evaluación de señales (s)	Instancias activas	Número de peticiones
5	15	1	200

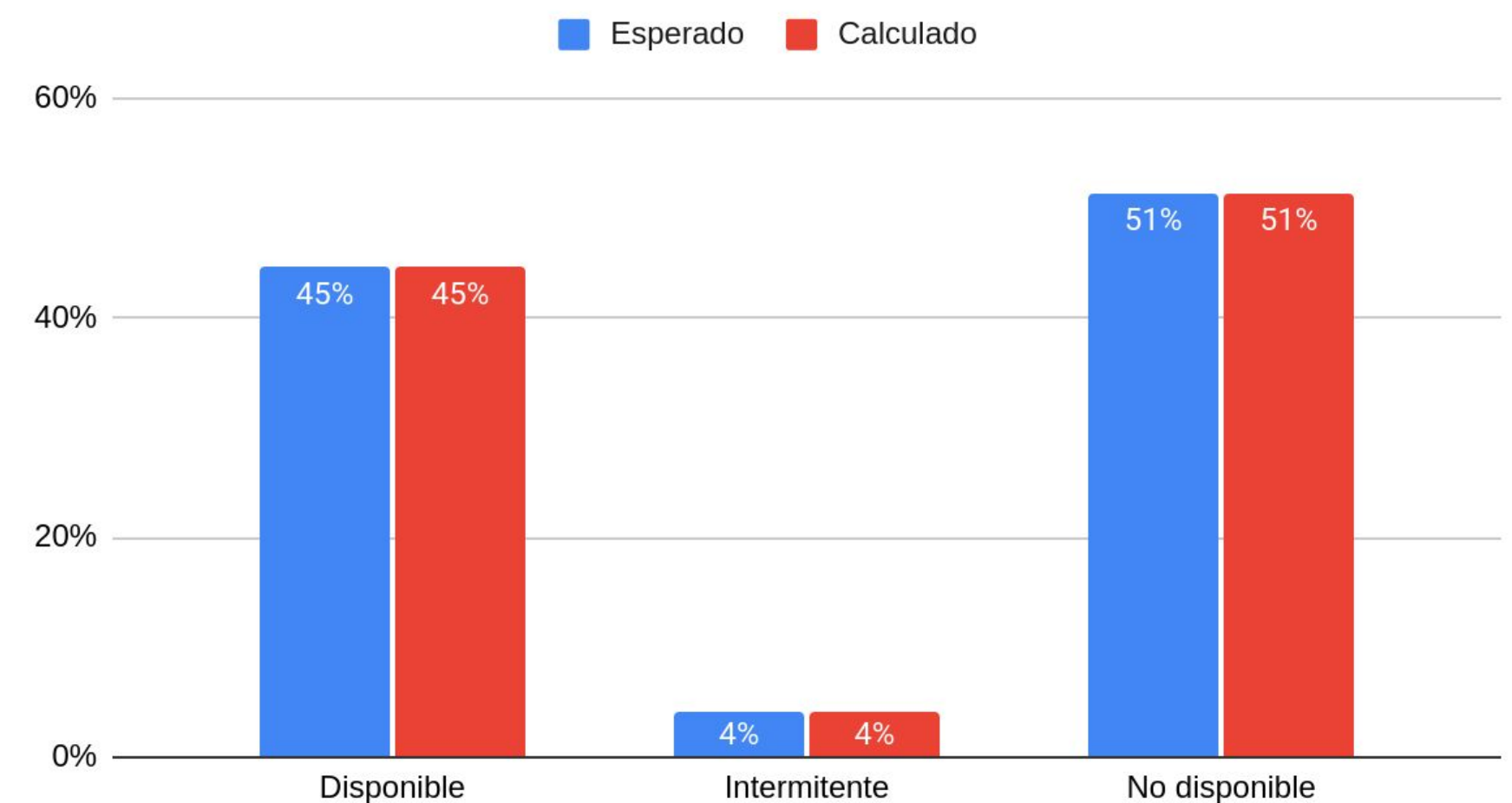
Flujo de la prueba

1. Registro de instancia
2. Envío 50 señales buenas
3. Envío 30 señales malas
4. Envío 10 señales buenas
5. Envío 30 señales malas
6. Envío 50 señales buenas
7. Envío 30 señales malas
8. shutdown de instancia

Indice de disponibilidad del servicio



Esperado Vs Calculado



Resultados - Escenario 3

Parametros

Periodicidad envío de señales (s)	Periodicidad evaluación de señales (s)	Instancias activas	Número de peticiones
5	30	2	50

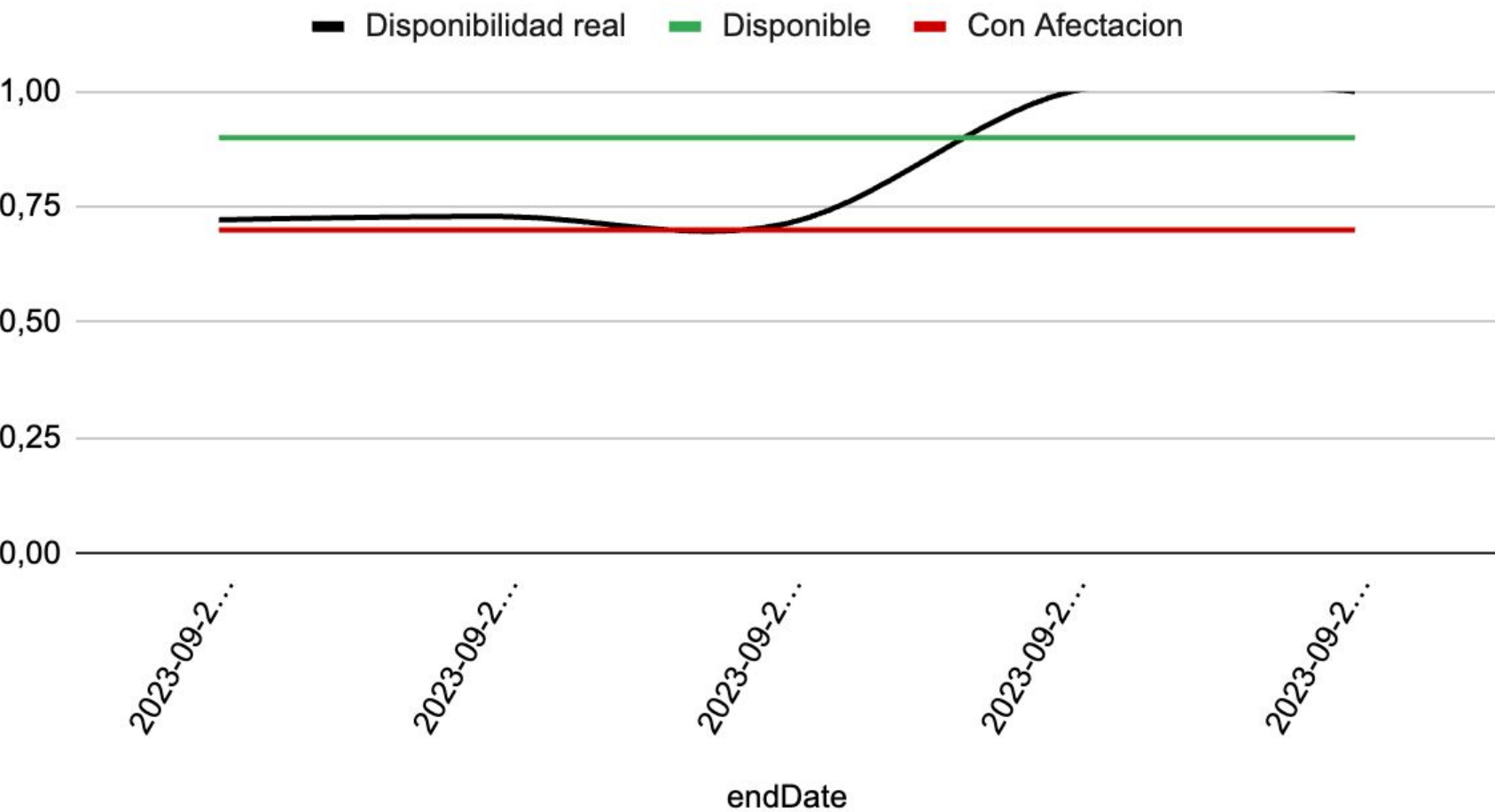
Flujo de la prueba instancia 1

- 1. Registro de instancia
- 2. Envío 10 señales buenas
- 3. Envío 5 señales malas
- 4. Envío 10 señales buenas
- 5. shutdown de instancia

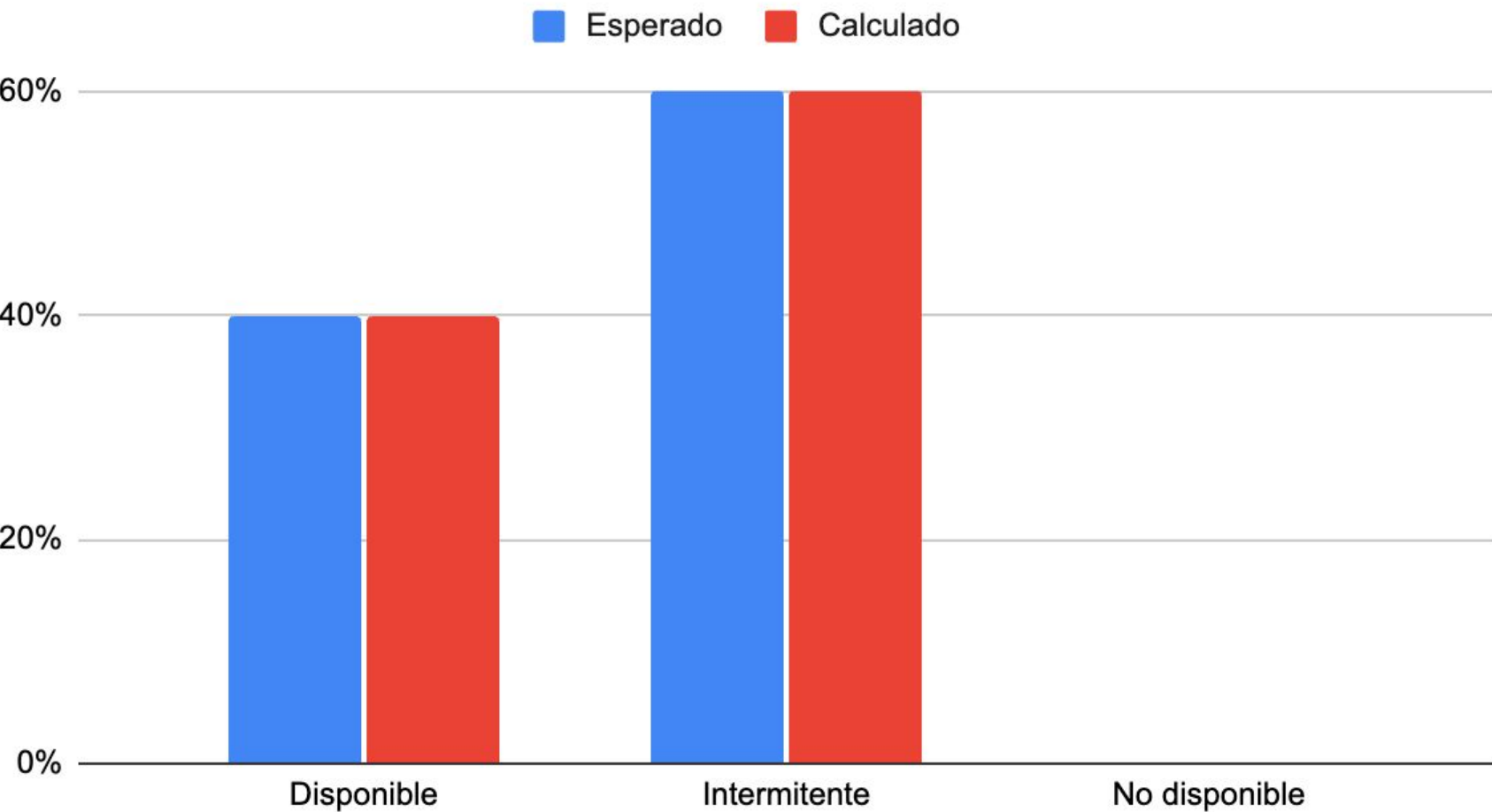
Flujo de la prueba instancia 2

- 1. Registro de instancia
- 2. Envío 10 señales buenas
- 3. Envío 5 señales malas
- 4. Envío 10 señales buenas
- 5. shutdown de instancia

Indice de disponibilidad del servicio



Esperado Vs Calculado



Resultados - Escenario 4

Parametros

Periodicidad envío de señales (s)	Periodicidad evaluación de señales (s)	Instancias activas	Número de peticiones
5	30	2	265

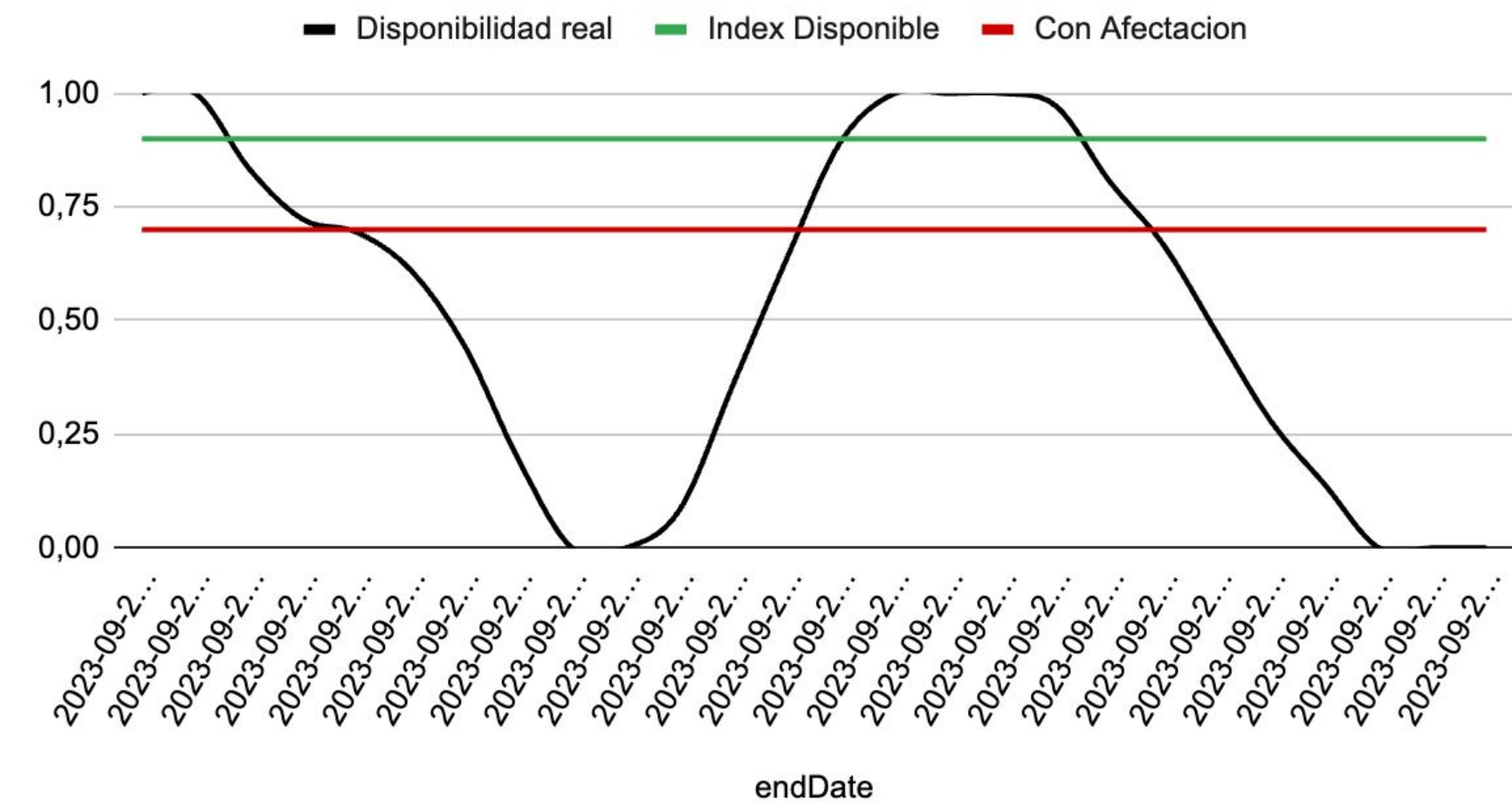
Flujo de la prueba instancia 1

1. Registro de instancia
2. Envío 10 señales buenas
3. Envío 5 señales malas
4. Envío 10 señales buenas
5. Envío 30 señales malas
6. Envío 40 señales buenas
7. Envío 30 señales malas
8. shutdown de instancia

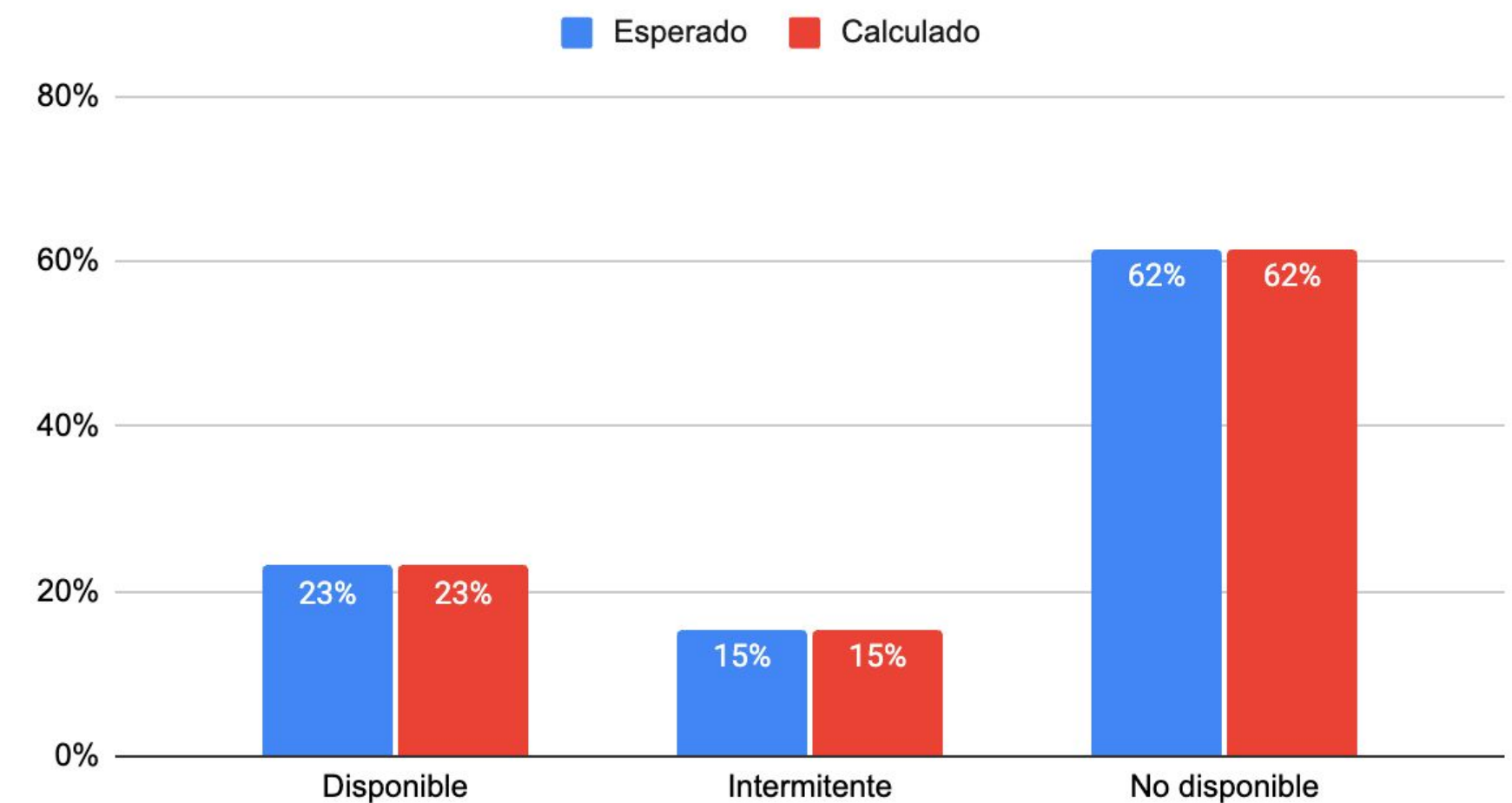
Flujo de la prueba instancia 2

1. Registro de instancia
2. Envío 15 señales buenas
3. Envío 5 señales malas
4. Envío 10 señales buenas
5. Envío 30 señales malas
6. Envío 50 señales buenas
7. Envío 30 señales malas
8. shutdown de instancia

Indice de disponibilidad del servicio



Esperado vs Calculado



Resultados - Escenario 5

Parametros

Periodicidad envío de señales (s)	Periodicidad evaluación de señales (s)	Instancias activas	Número de peticiones
5	15	2	265

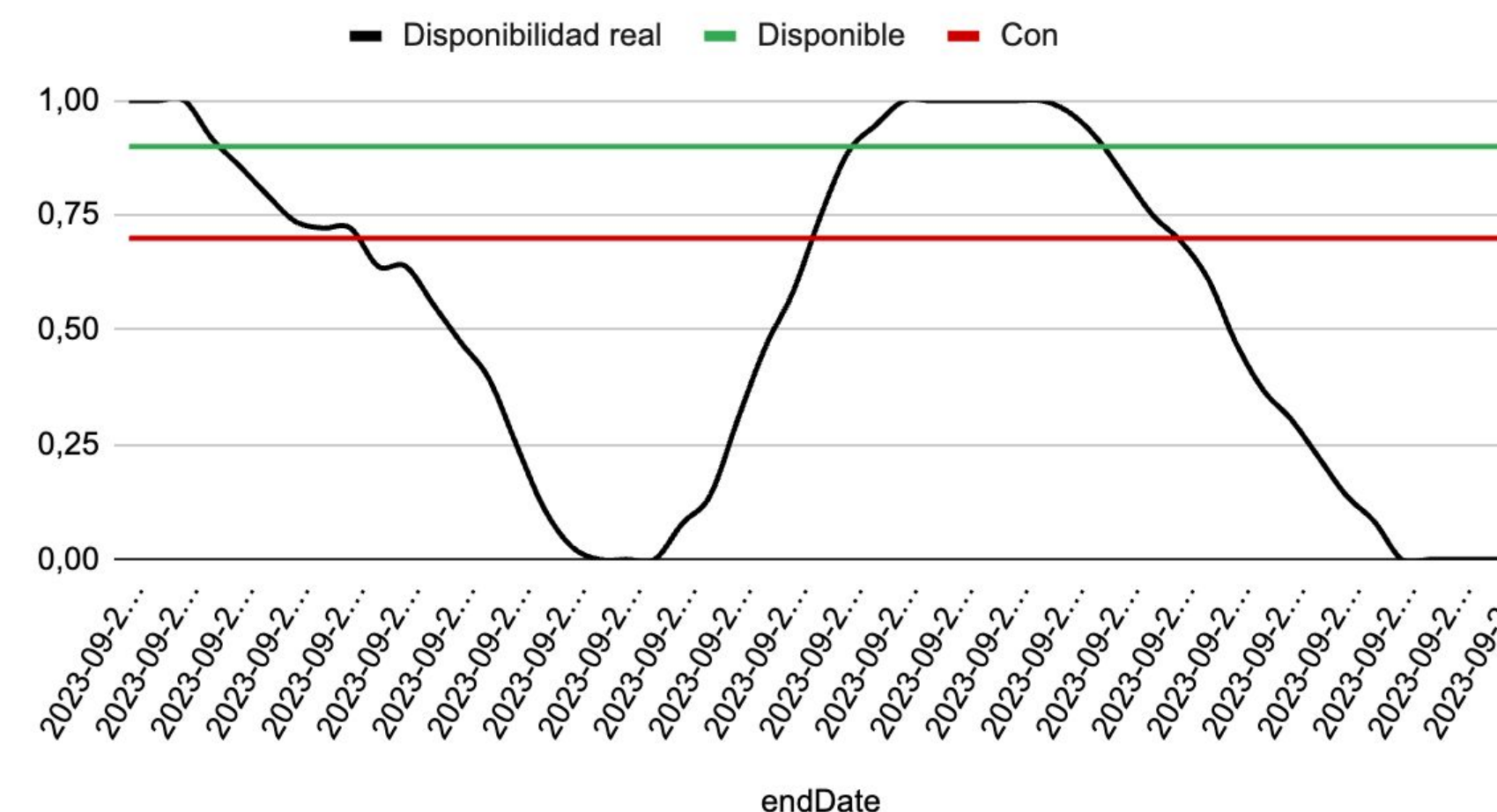
Flujo de la prueba instancia 1

1. Registro de instancia
2. Envío 10 señales buenas
3. Envío 5 señales malas
4. Envío 10 señales buenas
5. Envío 30 señales malas
6. Envío 40 señales buenas
7. Envío 30 señales malas
8. shutdown de instancia

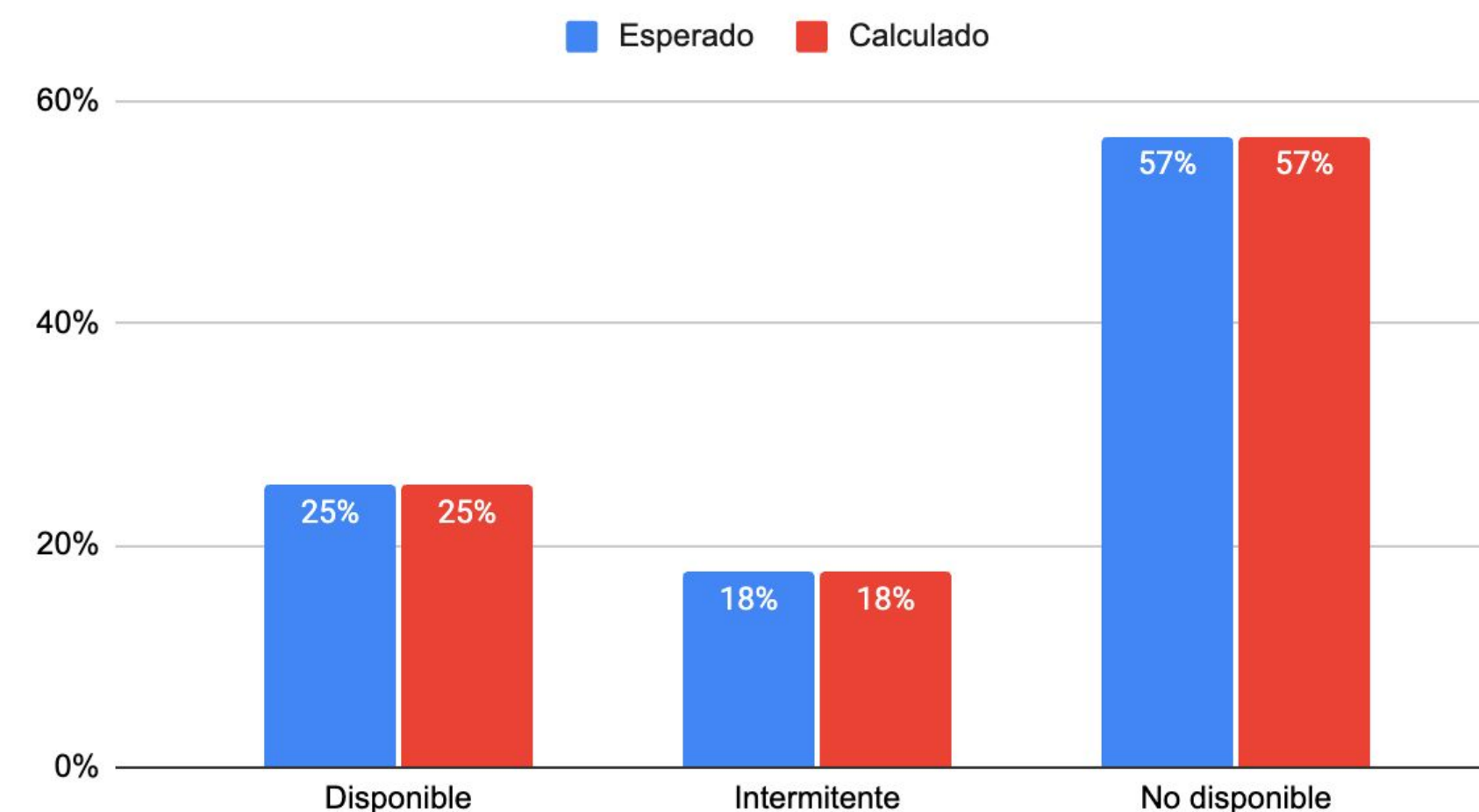
Flujo de la prueba instancia 2

1. Registro de instancia
2. Envío 15 señales buenas
3. Envío 5 señales malas
4. Envío 10 señales buenas
5. Envío 30 señales malas
6. Envío 50 señales buenas
7. Envío 30 señales malas
8. shutdown de instancia

Indice de disponibilidad del servicio



Esperado vs Calculado



Resultados - Orquestador

Parametros

Periodicidad d envío de señales (s)	Periodicidad d evaluación de señales (s)	Instancias activas	Número de peticiones	Periodicidad d de consulta (s)
5	15	2	100	5

Flujo de la prueba instancia 1

- 1. Registro de instancia
- 2. Envío 10 señales buenas
- 3. Envío 5 señales malas
- 4. Envío 10 señales buenas
- 5. Envío 30 señales malas
- 6. Envío 40 señales buenas
- 7. Envío 30 señales malas
- 8. shutdown de instancia

Flujo de la prueba instancia 2

- 1. Registro de instancia
- 2. Envío 15 señales buenas
- 3. Envío 5 señales malas
- 4. Envío 10 señales buenas
- 5. Envío 30 señales malas
- 6. Envío 50 señales buenas
- 7. Envío 30 señales malas
- 8. shutdown de instancia

Resultados

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput
evaluate-service	100	455	382	718	54.64	0%	18.497
TOTAL	100	455	382	718	54.64	0%	18.497

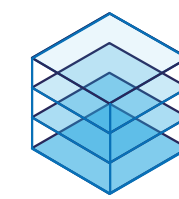
Análisis de los resultados obtenidos

- 1- Indique si la hipótesis de diseño pudo ser confirmada o no
- 2- En caso de que la hipótesis se haya confirmado, explique las decisiones de arquitectura que favorecieron el resultado
- 3- En caso de que los resultados del experimento no hayan sido favorables, explique por qué y cuáles cambios realizaría en el diseño

1. La hipótesis fue confirmada

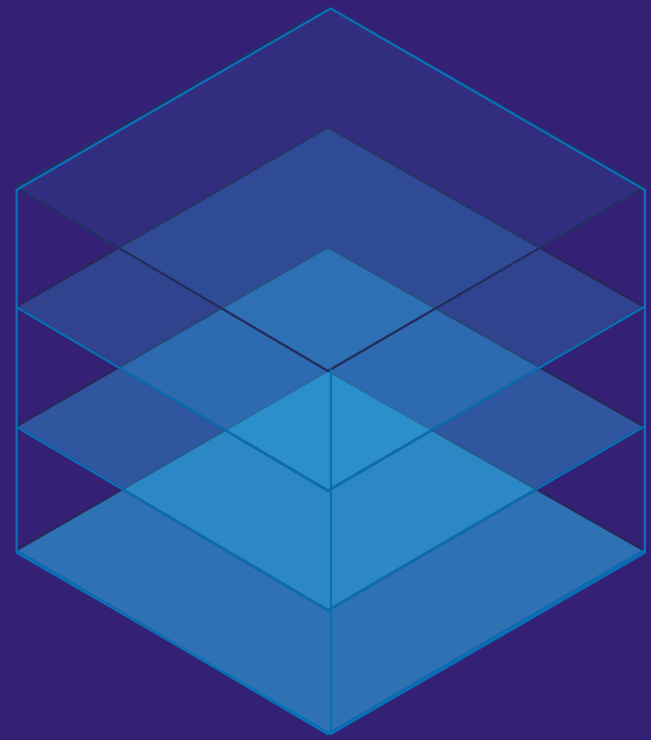
2. Dado que los resultados del experimento fueron favorables, se resaltan las decisiones de arquitectura que favorecieron el resultado:

- **Monitoreo Robusto:** Establecer un sistema sólido de monitoreo para rastrear el estado de salud del microservicio.
- **Señales Regulares:** Enviar peticiones de salud al broker de manera constante para verificar su funcionamiento.
- **Protocolo Confiable:** Utilizar un protocolo de comunicación confiable para los señales.
- **Manejo de Fallos:** Implementar mecanismos de enmascaramiento de respuestas para lidiar con problemas o fallos temporales.
- **Algoritmos de Evaluación:** Emplear algoritmos para analizar los datos de las señales y determinar si el microservicio funciona correctamente.
- **Escalabilidad:** Diseñar la arquitectura para que pueda monitorear múltiples microservicios de manera eficiente.
- **Pruebas Rigurosas:** Realizar pruebas exhaustivas para garantizar que el sistema cumpla con su objetivo del 99,9% de detección.
- **Supervisión Continua:** Monitorear el sistema en producción de forma continua y realizar ajustes según sea necesario para mantener su eficacia.



Conclusiones

- La hipótesis de diseño SI pudo ser confirmada, ya que se esperaba que el componente monitor identificara la caída y/o mal funcionamiento del microservicio el 99,9% de las veces para poder enmascarar el error al usuario y esto se logró en todos los escenarios de prueba realizados.
- La creación de un Índice de disponibilidad facilitó la identificación de los estados de los microservicios ya que por la cantidad de datos analizados es necesario procesarlos para tomar decisiones.
- Además de monitorear el estado binario de los microservicios UP/down se puede identificar si el microservicio presenta intermitencia o afectación en su funcionamiento.
- Además de los estados ya identificados y monitoreados se observa que se puede expandir a monitorear otras variables como; demoras en transacciones, bloqueos, diferencia de carga.
- Se establecieron los índices de disponibilidad de 0.9 y de intermitencia de 0.7 con el fin de determinar la disponibilidad del servicio y poder identificar si se encuentra en correcto funcionamiento, si presenta intermitencia o si presenta indisponibilidad
- De acuerdo a los escenarios ejecutados, el intervalo de tiempo adecuado para el envío de señales heartbeat es 5 segundos
- De acuerdo a los escenarios ejecutados, el tiempo adecuado de revisión de estado de las instancias de los microservicios es 15 segundos.



MISO

Maestría en Ingeniería de
Software

Gracias