**PLAN DE PRUEBAS DE PERFORMANCE**

El siguiente documento tiene como objetivo brindar el detalle de la estrategia de las pruebas de performance a realizar para el squad “*Framework de Desarrollo Backend*”. Se evaluará la información de entrada de la funcionalidad/API *“Migración Message-Hub OCP 4.8”*, abordando los criterios de riesgo que definirá el plan de las pruebas de performance.

Se detallará la viabilidad técnica, las funcionalidades y sus criterios de aceptación, la arquitectura de componentes y servidores. También de definirá los tipos de pruebas de performance a realizar en base a los riesgos de negocio y técnicos, y por último la guía de monitoreo que se deberá seguir durante la ejecución de las pruebas.

**Tabla N°01:** *Versión del documento Análisis de Pruebas de Performance*

|  |  |  |  |
| --- | --- | --- | --- |
| Versión | Descripción del cambio | Autor | Fecha de Creación |
| *V 1.0* | *Creación de Plan de Pruebas.* | *Cindy John* | *03-10-2022* |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contenido

[**1. ENTENDIMIENTO DE FUNCIONALIDAD/API A NIVEL DE NEGOCIO** 4](#_Toc115707227)

[**1.1** **Perfiles del equipo de trabajo** 4](#_Toc115707228)

[**1.2. Viabilidad Técnica** 4](#_Toc115707229)

[**1.2.1.** **Checklist de performance:** 4](#_Toc115707230)

[**2. ESTRATEGIA** 5](#_Toc115707231)

[2.1. **Definición e identificación de objetivos y criterios de aceptación** 5](#_Toc115707234)

[**2.2.** **Entendimiento de la arquitectura física** 6](#_Toc115707235)

[***2.2.1.*** **Framework que validar.** 6](#_Toc115707236)

[**2.3.** **Determinar tipos de performance testing basado en riesgos** 7](#_Toc115707237)

[**2.3.1.** **Identificación de riesgos** 7](#_Toc115707238)

[**2.4.** **Escenarios de performance testing** 7](#_Toc115707239)

[**2.5.** **Monitoreo** 8](#_Toc115707240)

[**3.** **CRONOGRAMA DEL PROYECTO *(Sección Obligatoria)*** 9](#_Toc115707241)

[**4.** **CONSIDERACIONES DEL PROYECTO** 9](#_Toc115707242)

# **1. ENTENDIMIENTO DE FUNCIONALIDAD/API A NIVEL DE NEGOCIO**

Esta opción consiste en migrar el API message-hub de openshift 3.11 a openshift 4.8.

La funcionalidad será la misma.

## **Perfiles del equipo de trabajo**

**Tabla N°02:** *Equipo de trabajo*

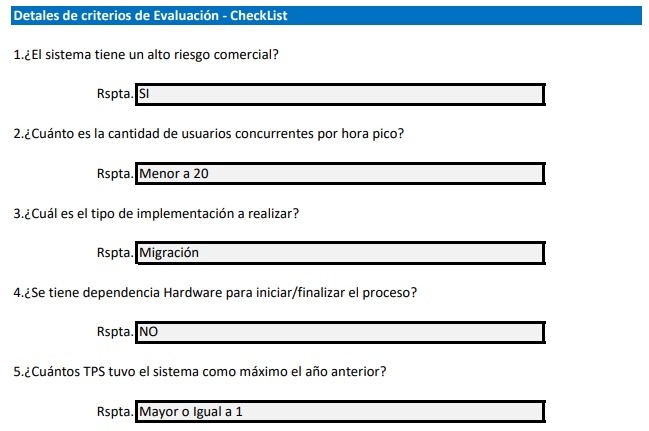
|  |  |
| --- | --- |
| **indicar perfil/cargo** | ATLAS |
| **Chapter Lead** | Francisco Alejandro André |
| **Líder Técnico** | Jose Luis Vargas |
| **QA** | Cindy John |
| **Desarrollador (starter)** | Pierre Obregon |

## **1.2. Viabilidad Técnica**

Para determinar la viabilidad de las pruebas de performance para la funcionalidad *“Migración Message-Hub OCP 4.8”*, se ha utilizado el artefacto “*Criterios de identificación para Performance Testing*” donde se obtuvo los siguientes resultados:

A continuación, se indica los criterios encontrados:

## **Checklist de performance:**





Adjunto N° 1: Documento que sustenta la viabilidad técnica y negocio

# **2. ESTRATEGIA**

En esta sección se verá el paso a paso de la Estrategia a utilizar para el entendimiento de la “*Migración Message-Hub OCP 4.8*” y el abordamiento de los riesgos identificados con el artefacto Matriz de Riesgos, considerando los tipos de pruebas de performance propuestos por el artefacto, así como también la guía de monitoreo que se deberá seguir en la ejecución de las pruebas de performance.



## **Definición e identificación de objetivos y criterios de aceptación**

* Objetivos de usuario y Técnicos *Migración Message-Hub OCP 4.8*:

Para la identificación de los objetivos y criterios de aceptación, se ha obtenido los objetivos de usuario y los objetivos técnicos.

* Número de usuarios concurrentes esperado por cada endpoint.
* Número de transacciones por segundo esperado por cada endpoint.
* Tiempo de respuesta esperado por petición en cada uno de los endpoint.
* Porcentaje de error menor o igual a 1%
* Pruebas con 1/4 Pod en ambiente de certificación.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **OCP** | | | | |
| Endpoint | Concurrencia | Tiempo promedio de respuesta (Segundos) | Transacciones por segundo (TPS) | Porcentaje de Error |
| send | 650 | <=1 seg. | >=18/s | <=0.5% |
| sendAsyncr | 650 | <=1 seg. | >=18/s | <=0.5% |

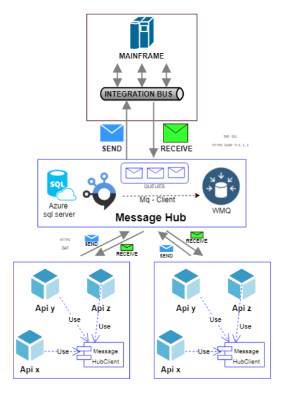
## **Entendimiento de la arquitectura física**

El api para evaluar pasa las funcionalidades:

* Message-Hub

## **Framework que validar.**

Diagrama de Arquitectura Message-Hub

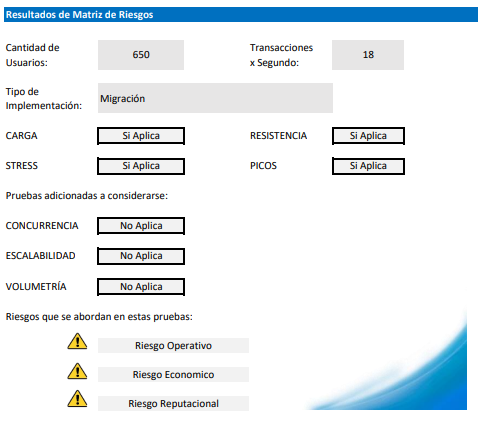


El Message hub es un api de soporte, el cual te permite comunicarte con el HOST para ejecutar los programas que residen en él y enviar el resultado de dichos programas como respuesta a la invocación de este api. Para esto expone un api rest, el cual envía mensajes a colas MQ para ser consumidas por un flujo de bróker (el cual ejecuta llama a la ejecución de dicho programa enHOST) o por los programas con adaptador HOST, las respuestas de estos programas son enviados a colas para ser consumidas por este microservicio message hub

## **Determinar tipos de performance testing basado en riesgos**

En esta sección se identifica los riesgos que impactan al “Migración Message-Hub OCP 4.8” para ello se utiliza el artefacto Matriz de Riesgos, el cual nos indica los riesgos que se abordan en estas pruebas, así como las pruebas de performance obligatorias.

## **Identificación de riesgos**



Para mayor detalle se adjunta el artefacto “*Matriz de Riesgos de Performance Testing*”.



## **Escenarios de performance testing**

En esta sección se indica los escenarios diseñados para las pruebas de Carga, Estrés, Picos y Resistencia para el “Migración Message-Hub OCP 4.8”. Se ha utilizado el artefacto “Diseño de escenarios de Performance Testing” para el diseño de los casos.



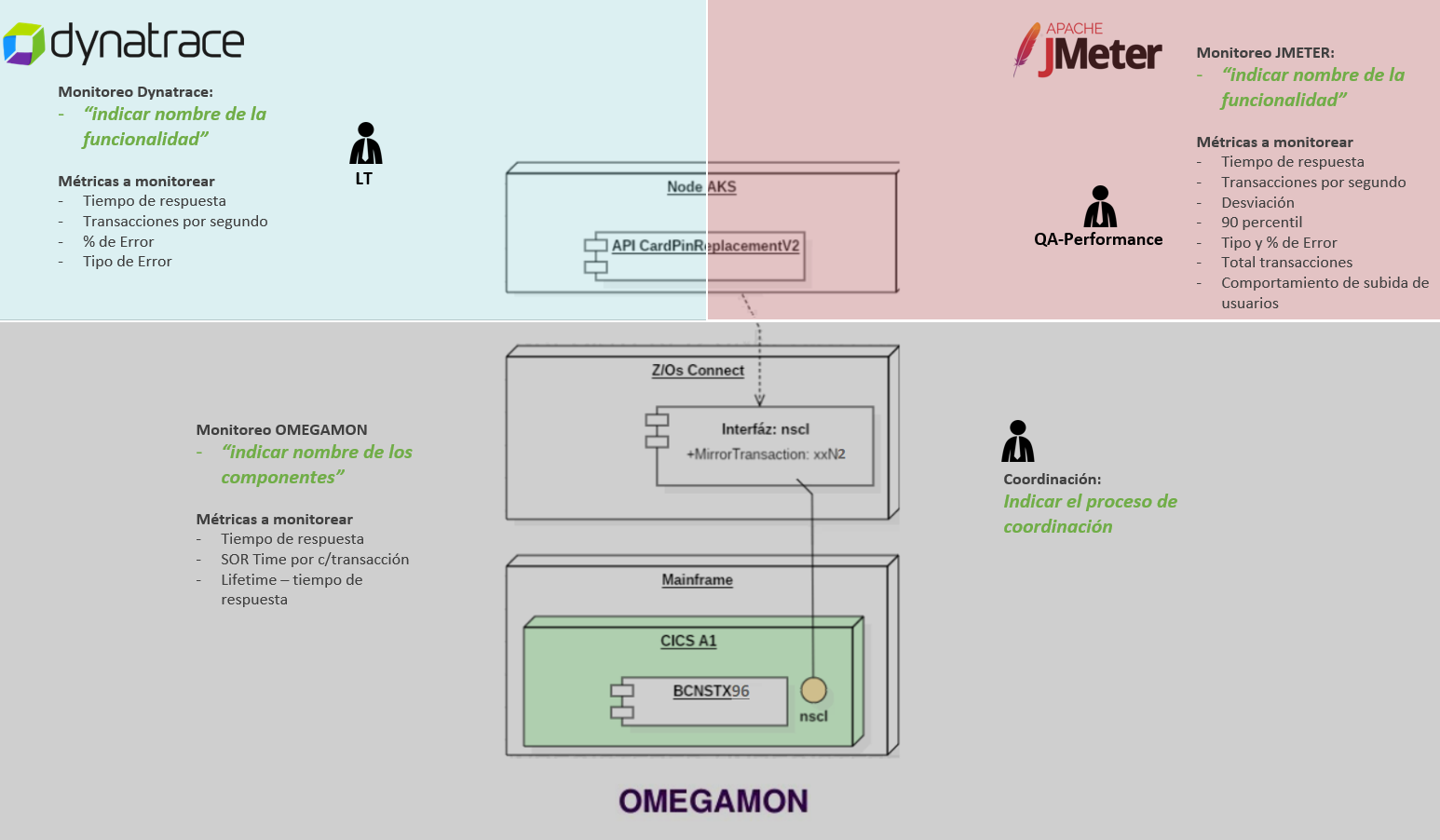
* + 1. Escenarios

|  |  |  |
| --- | --- | --- |
| **Tipo de Prueba** | **Escenario** | **Configuracion** |
| Stress |  |  |
| Para las funcionalidades:  • Message-Hub (OCP)  **Dado** que 2500 usuarios quieren realizar una consulta al endpoint del api de error-catalog.  **Cuando** envían su solicitud - request  **Entonces** obtiene código de respuesta 200  **Y** en tiempo promedio de respuesta es menor o igual a 2 segundos  **Y** las transacciones por segundo mayor o igual a 400TPS  **Y** un porcentaje de error menor o igual a 1% | La inyección de usuarios consistirá en pruebas escalonadas, iniciando con 100 usuarios concurrentes para los primeros 5 minutos, luego subirán 300 usuarios concurrentes para los siguientes 5 minutos, así sucesivamente hasta llegar a un total de 2500 usuarios concurrentes, durante un total de 30 minutos aproximadamente, de acuerdo con la imagen del diseño de escenarios de performance testing. Para más detalle se adjunta el artefacto “Diseño de escenarios de Performance Testing” |
| Carga |  |  |
| Para las funcionalidades:  • Message-Hub (OCP)  **Dado** que 650 usuarios quieren realizar una consulta al endpoint del api de error-catalog.  **Cuando** envían su solicitud - request  **Entonces** obtiene código de respuesta 200  **Y** en tiempo promedio de respuesta es menor o igual a 1 segundos  **Y** las transacciones por segundo mayor o igual a 18TPS  **Y** un porcentaje de error menor o igual a 0.5% | La inyección de usuarios consistirá en pruebas escalonadas, iniciando con 50 usuarios concurrentes para los primeros 5 minutos, luego subirán 150 usuarios concurrentes para los siguientes 5 minutos, así sucesivamente hasta llegar a un total de 650 usuarios concurrentes, durante un total de 30 minutos aproximadamente, de acuerdo con la imagen del diseño de escenarios de performance testing. Para más detalle se adjunta el artefacto “Diseño de escenarios de Performance Testing” |
| Picos |  |  |
| Para las funcionalidades:  • Message-Hub (OCP)  **Dado** que 2000 usuarios quieren realizar una consulta al endpoint del api de error-catalog.  **Cuando** envían su solicitud - request  **Entonces** obtiene código de respuesta 200  **Y** en tiempo promedio de respuesta es menor o igual a 1 segundos  **Y** las transacciones por segundo mayor o igual a 400TPS  **Y** un porcentaje de error menor o igual a 1% | La inyección de usuarios consistirá en pruebas escalonadas, iniciando con 100 usuarios concurrentes para los primeros 5 minutos, luego subirán 1500 usuarios concurrentes para los siguientes 5 minutos, posteriormente bajará a 300 usuarios, luego subirá a 2000 usuarios, para finalmente bajar a 200 usuarios, durante un total de 30 minutos aproximadamente, de acuerdo con la imagen del diseño de escenarios de performance testing. Para más detalle se adjunta el artefacto “Diseño de escenarios de Performance Testing” |

## **Monitoreo**

Se realizará el monitoreo desde Dynatrace y Apache Jmeter

**Ilustración N°04** – Monitoreo de componentes por herramienta



# **CRONOGRAMA DEL PROYECTO *(Sección Obligatoria)***

**Tabla N° 05 –** Cronograma de actividades

|  |  |  |  |
| --- | --- | --- | --- |
| **Fase** | **Sprint 6** | **Equipo** | **Responsable** |
| 28/09 **al 11/10** | **Soporte** |
| Preparación |  |  | QA |
| Estrategia |  |  | QA |
| Implementación |  |  | QA |
| Configuración |  | ATLAS | QA |
| Ejecución y Analítica |  | ATLAS | QA |

*\* Los tiempos de ejecución pueden variar dependiendo de la finalización de las pruebas funcionales.*

# **CONSIDERACIONES DEL PROYECTO**

* Se ha considerado que los accesos al ambiente de certificación.
* Durante la etapa de análisis se requiere coordinaciones técnicas con el Equipo de Proyecto.
* Se requiere apoyo del LT o Desarrollador para el monitoreo de Dynatrace.
* Se realizará pruebas de performance de los EndPoint correspondientes a la versión del api de soporte message-hub ocp 4.8.