PROYECTO DE IMPLEMENTACION CENTRADO EN ARQUITECTURA - PICA

“TouresBalón”

Entrega 1 - Diseño

RODRIGO BASTIDAS MARTINEZ

RIGOBERTO GIRALDO CARMONA

LUZ EDITH GONZALEZ PALENCIA

NATALIA LEON CARDOZO

CARLOS ARTURO MERCHAN

Tabla de contenido

[**1.** **Introducción** 3](#_Toc53250104)

[**2.** **Problema para resolver** 3](#_Toc53250105)

[**3.** **Alcance** 3](#_Toc53250106)

[**4.** **Vistas de Arquitectura** 4](#_Toc53250107)

[**4.1.** **Vista de Casos de Uso** 4](#_Toc53250108)

[**4.2.** **Vista de Componentes** 4](#_Toc53250109)

[**4.3.** **Vista de Despliegue** 5](#_Toc53250110)

[**5.** **Justificación de diseño y tecnologías** 6](#_Toc53250111)

[**6.** **Escenarios de negocio** 7](#_Toc53250112)

[**6.1.** **Registrar un nuevo proveedor** 7](#_Toc53250113)

[**6.2.** **Retirar un proveedor de TouresBalon** 8](#_Toc53250114)

[**6.3.** **Consultar cotizar espectáculo** 9](#_Toc53250115)

[**7.** **Decisiones de Arquitectura** 10](#_Toc53250116)

[**8.** **Patrones de arquitectura** 10](#_Toc53250117)

## **Introducción**

El presente documento complementa el planteamiento de la arquitectura para la implementación de la integración de proveedores que prestarán servicios de transporte, hospedaje y espectáculos deportivos a la empresa de eventos deportivos TouresBalón

## **Problema para resolver**

TouresBalón requiere que:

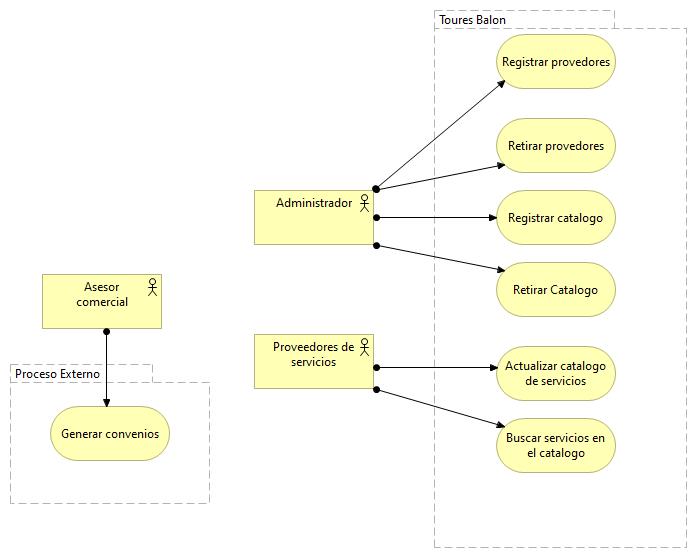
1. La implementación de los proveedores y su interacción con el proceso sea desacoplada. Esto apoyado por servicios federados que interactúen como abstracciones para el proceso.
2. Se integren nuevos proveedores con el menor impacto para la arquitectura y sus componentes, limitando los componentes que deben verse modificados. Preferiblemente un nuevo proveedor debería ser plug & play para tecnologías que ya son soportadas.
3. Dependiendo de la negociación con el proveedor, estos pueden tener un mayor peso cuando se está realizando las búsquedas de opciones, de manera que se puedan ofrecer como los preferidos. Esos valores pueden cambiar dinámicamente dependiendo de nuevas condiciones de negocio.
4. La forma en que se debe mostrar los resultados debe tener en cuenta: Peso del convenio 50% (1-10) + Calificación de los usuarios 50% (1-10). Los porcentajes pueden cambiar con el tiempo.
5. La comunicación con los proveedores puede tardar algunos segundos, así que se espera que se pueda realizar una comunicación en paralelo para poder así aprovechar mejor los recursos de cómputo y también como manejar los fallos.

## **Alcance**

El alcance de este taller corresponde a presentar los diagramas necesarios para resolver el problema planteado por TouresBalón y expresar las decisiones que se tomaron para la arquitectura justificando el cómo y el porqué de las decisiones tomadas

## **Vistas de Arquitectura**

## **Vista de Casos de Uso**

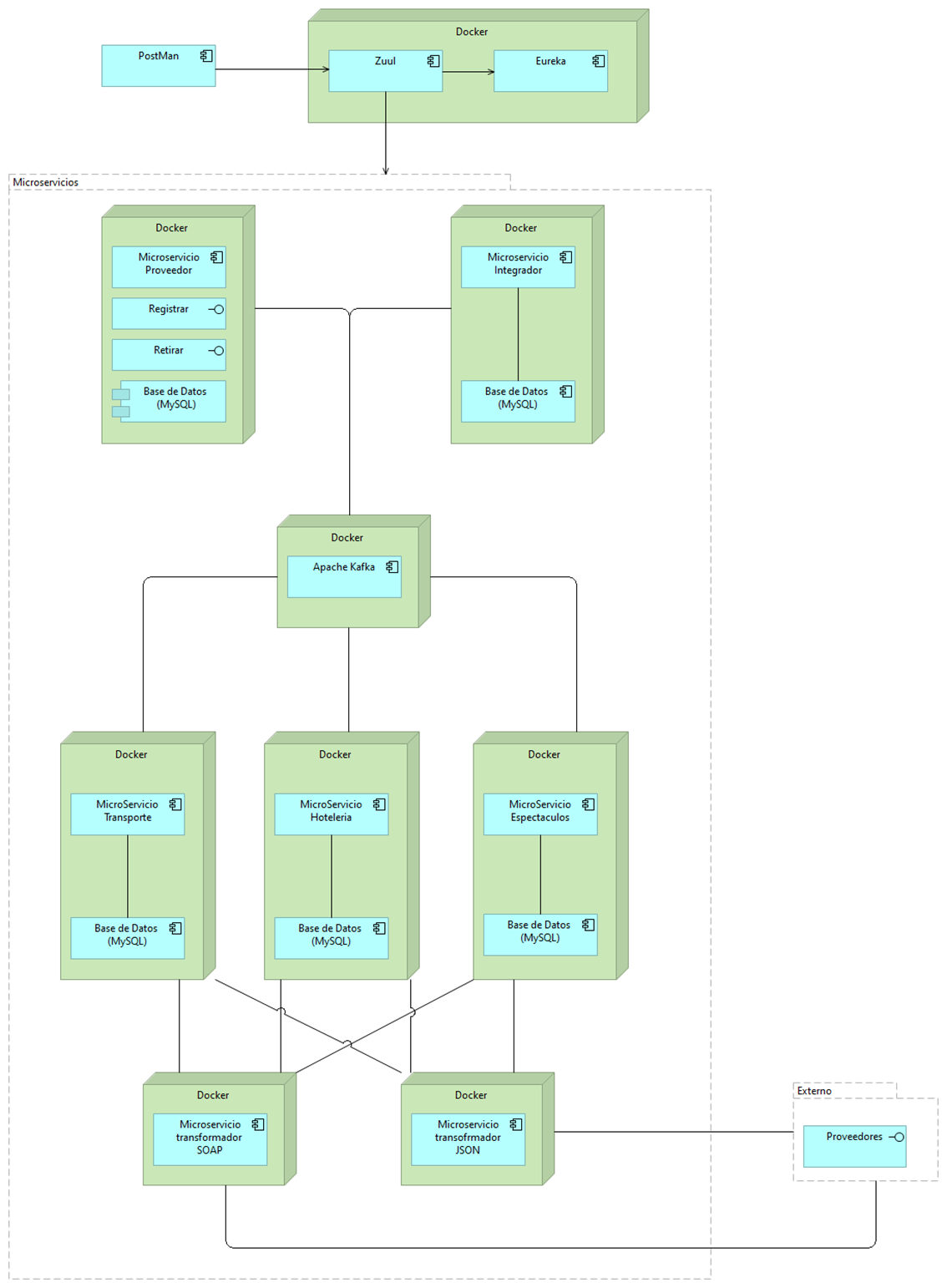


## **Vista de Componentes**

Diagrama

Descripción generada automáticamente

## **Vista de Despliegue**

.

## **Justificación de diseño y tecnologías**

La solución de arquitectura planteada cumple con los requerimientos de negocio. Está orientada a servicios y eventos, sobre los cuales se gestionará la información relacionada al registro de los proveedores, cotización y reservas que confirman los servicios ofrecidos por TouresBalon como son hotelería, transporte y hospedaje. El proceso de integración con los proveedores debe ser en línea ya que pueden presentarse una variación de proveedores o de los servicios ofrecidos por cada uno ellos. A continuación, se describen los componentes fundamentales que soportan la arquitectura

| **Componente** | **Descripción** |
| --- | --- |
| POSTMAN | Plataforma que permite trabajar directamente con las APIs expuestas sin necesidad de contar con una interfaz de usuario. Es un componente temporal mientras se implementa la Interfaz de usuario. |
| ZUUL | API Gateway. Es el componente encargado de exponer uniformemente los servicios que serán accesibles desde el exterior y gestionar el acceso a los diferentes microservicios. |
| EUREKA | Componente encargado de registrar los microservicios de la aplicación que serán utilizados en el proceso de integración de los nuevos proveedores de cada área. Ofrece funciones de descubrimiento. |
| REDIS CACHE | Componente de caché que almacena información con poca variabilidad y de constante consulta, por ejemplo, información asociada a la configuración de las plantillas de cada proveedor por servicio y por operación (mapeo). |
| APACHE KAFKA | componente de transmisión de datos por medio de eventos bajo el modelo publicador/suscriptor. Apoya la coreografía de microservicios. |
| MICROSERVICIO PROVEEDOR | Es el componente que tiene como función realizar el registro de un nuevo proveedor, guardando los convenios de cada uno, así como los datos relacionados con el acceso a los puntos de integración, también permitiendo dar de baja aquellos proveedores que decidan retirar sus servicios del sistema de TouresBalón. |
| MICROSERVICIO INTEGRADOR | Es el componente encargado de recibir y consolidar las solicitudes de cotización sobre los servicios ofrecidos por TouresBalón, apoyado mediante composición de los microservicios de hotelería, transporte y espectáculo. |
| MICROSERVICIO TRANSPORTE | componente encargado de atender las solicitudes de información de servicios de transporte necesarios para llevar a cabo los procesos de negocio tales como cotización o reserva. |
| MICROSERVICIO HOTELERIA | componente encargado de atender las solicitudes de información de servicios hoteleros necesarias para llevar a cabo los procesos de negocio tales como cotización o reserva. |
| MICROSERVICIO ESPECTÁCULO | componente encargado de atender las solicitudes de información de espectáculos necesarias para llevar a cabo los procesos de negocio tales como cotización o compra. |
| MICROSERVICIO TRANSFORMADOR SOAP | Este microservicio se encarga de obtener la información de los servicios SOAP ofrecidos de un proveedor basándose en los datos previamente guardados, aplicando los métodos y transformaciones necesarias para leer los datos obtenidos y comunicarlos al microservicio de transporte, hotelería o espectáculo asociado |
| MICROSERVICIO TRANSFORMADOR JSON | Tiene la responsabilidad de comunicarse con los servicios REST/JSON ofrecidos por los proveedores utilizando la parametrización definida para cada convenio y realizando las transformaciones necesarias para interpretar la información retornada de la comunicación con cada uno de ellos y mapeándola a cada proceso que corresponda sobre los servicios ofrecidos por TouresBalón en hotelería, transporte y hospedaje |

## **Escenarios de negocio**

## **Registrar un nuevo proveedor**

**Paso 1**: El Asesor comercial de Toures Balón firma convenio con el proveedor. El contrato debe tener un anexo técnico donde indique las características de la interfaz que permita correlacionar las operaciones de TouresBalon con las operaciones del sistema proveedor.

Las operaciones requeridas son:

* Consultar datos de proveedor
* Consultar catálogo de servicios de proveedor
* Cotizar un servicio especifico
* Reservar
* Confirmar reserva
* Cancelar reserva

El anexo técnico también define los endpoints donde se expondrán las interfaces de dicho proveedor, así como el formato y protocolo de comunicación.

**Paso 2**: El asesor registra los datos del proveedor en el sistema acorde al convenio establecido. Para efectos del taller se utilizará Postman para invocar el API del microservicio del proveedor que ofrece esta funcionalidad. En este punto también se registra el peso del convenio.

**Paso 3**: El asesor registra el mapeo de datos referente a operaciones para extraer la información de los servicios del proveedor y transformar estos datos al modelo canónico de TouresBalon. Para efectos de este taller se hará directamente sobre la base de datos del microservicio - Proveedor.

La integración de proveedores de hotelería, transporte y hospedaje comienza con el registro de los convenios y la parametrización que permita identificar el tipo de integración y los requisitos que se deben cumplir para invocar los servicios expuestos por cada proveedor, esta información quedará registrada sobre la base de datos asociada al microservicio proveedor

## **Retirar un proveedor de TouresBalon**

Paso 1: Postman PUT /proveedor/{idProveedor}. Esta acción es atendida por el microservicio: Proveedor e inactiva un proveedor (borrado lógico).

## **Consultar cotizar espectáculo**

**Paso1:** Se realiza la petición mediante Postman al microservicio integrador. La petición recibe una cadena de texto que funciona como filtro para búsqueda de espectáculos registrados en TouresBalón.

**Paso 2**: La anterior solicitud es atendida inicialmente por el microservicio integrador, quien emite un mensaje a Kafka para lograr comunicación con el microservicio de proveedores y así conocer los proveedores activos de espectáculos.

**Paso 3:** El microservicio de proveedor escucha el mensaje y emite el listado de proveedores activos para que sean procesados por el microservicio de espectáculos.

**Paso 4**: Para que el microservicio de espectáculos pueda atender la solicitud debe estar suscrito al tópico correspondiente en Kafka e invocar el conector/transformador correspondiente para cada proveedor.

**Paso 5:** Se realiza la conexión con la interfaz parametrizada por el proveedor del servicio y se realizan las transformaciones al mensaje entregado acorde al mapeo registrado. La respuesta es retornada por el servicio de espectáculo quien emite un evento a Kafka en el tópico correspondiente para que las respuestas sean finalmente consolidadas por el microservicio integrador.

**Paso 6**: A partir de los resultados de búsqueda, se simula la selección de un espectáculo por parte del usuario mediante Postman, el cual una emite una nueva petición de consulta del detalle sobre un espectáculo en particular, para así conocer los servicios de transporte y hospedaje relacionados.

**Paso 7:** Mediante Postman se simula la opción de conocer los proveedores de transporte. En este caso la solicitud es emitida por el componente integrador, pero es atendida por el microservicio de transporte, quien de manera similar invocará al componente conector/transformador para ubicar la interfaz del proveedor de servicios de transporte y obtener el catálogo de servicios relacionado al espectáculo.

**Paso 8:** Mediante Postman se simula la opción de conocer los proveedores de hospedaje. En este caso la solicitud es emita por el componente integrador, pero es atendida por el microservicio de hospedaje, quien de manera similar invocará al componente conector/transformador para ubicar la interfaz del proveedor de servicios de hospedaje y obtener el catálogo de servicios relacionado al espectáculo.

**Paso 9:** Los resultados de transporte y hospedaje son retornados al servicio integrador para presentar una cotización con el paquete completo (espectáculo, transporte y hospedaje)

## **Decisiones de Arquitectura**

* Plantear un estilo de arquitectura orientada a servicios teniendo en cuenta la solicitud de integración en línea con diferentes proveedores.
* Implementar y definir una arquitectura flexible que soporte los procesos de adición, eliminación y modificación de los proveedores sin que la integración de estos sea impactada.
* Implementar el uso de caché para guardar información que no varía constantemente como por ejemplo la parametrización de invocación de los proveedores. Sin embargo, por temas de consistencia de los datos la información de precios y disponibilidad de los servicios se consultará en línea directamente con el proveedor que ofrece el servicio.
* Como apoyo al gobierno de servicios, usar Eureka para descubrir y registrar los microservicios de la aplicación.
* Usar Zuul como API Gateway para centralizar y balancear el acceso hacia los microservicios en un solo punto.

## **Patrones de arquitectura empleados**

* Publish and Subscribe: Aplicado al uso de Kafka como el encargado de coreografiar los microservicios
* Event Message: Comunicación entre los microservicios mediante generación de eventos.
* Message Translator: Interpretar los mensajes de respuesta que se obtiene de los proveedores de TouresBalon.
* Agreggator: Unificación de mensajes entregados por los proveedores para generar una sola respuesta.
* Scatter Gatter: Envío de mensajes de manera asíncrona a los proveedores solicitados para generar una sola repuesta por cada proceso de negocio solicitado.
* Data Canonical Model: Establece un modelo uniforme de almacenamiento de los datos.
* Messaging Gateway: Centralizar y balancear el acceso hacia los microservicios en un solo punto.