

Integración de Aplicaciones

**Grado de Ingeniería
Informática**



Tercera entrega: Integración con Colas de Mensajería

**Jorge Gaitán Varea
Pablo Cerdá Puche
Jorge Mañanes Alfonso**

Curso 2023/2024



**UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA**

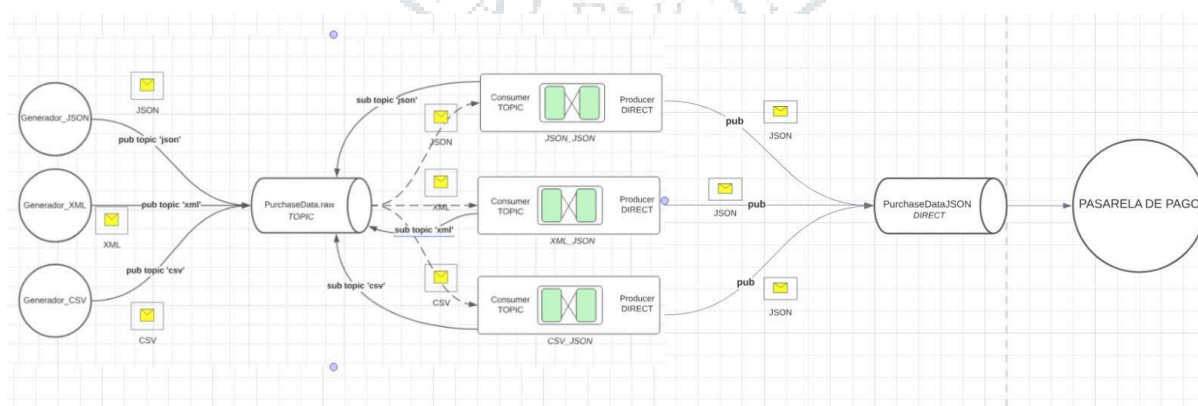
1. Descripción de la necesidad de integración

La plataforma Orienta-T se enfrenta a un desafío significativo al manejar las transacciones financieras de manera eficiente y segura. Los estudiantes utilizan la aplicación para acceder a recursos educativos que requieren de pagos, como cursos y materiales de estudio. La variedad de métodos de pago (tarjeta de crédito, PayPal, etc.) y la necesidad de un procesamiento de pagos seguro y en tiempo real exigen una solución confiable de integración.

Para llevar a cabo la resolución, conectaremos la API de Compras, que maneja las solicitudes de compra del usuarios con la Pasarela de Pago, que procesa dichas transacciones financieras. Posteriormente, debe asegurar que la confirmación de la transacción sea comunicada de vuelta a la API de Compras y al usuario final.

2. Diseño de la solución basada en colas

Nuestra solución utiliza una arquitectura de mensajería avanzada dentro de RabbitMQ, aprovechando tanto exchanges de tipo "topic" como "direct" para manejar eficientemente el flujo de datos y transacciones.



Flujo de Mensajes y Transformación de Datos

1. Generación de mensajes en diversos formatos:

Los mensajes se generan en tres diferentes formatos de datos (JSON, XML y CSV) por tres generadores distintos, reflejando la variedad de fuentes y tipos de datos que nuestra plataforma puede recibir.

2. Enrutamiento a través de Exchange "Topic":

Estos mensajes son inicialmente publicados al exchange "topic" PurchaseData.raw, que actúa como un colector de mensajes de múltiples formatos.

Utilizamos la capacidad de enrutamiento flexible de un exchange "topic" para dirigir los mensajes a las colas correspondientes basadas en su formato, utilizando claves de enrutamiento como 'json', 'xml' y 'csv'.

3. Transformación por Consumidores Específicos:

Cada cola tiene un consumidor dedicado que está suscrito a su respectivo topic. Estos consumidores son responsables de transformar los datos al formato JSON, que será el formato canónico para el procesamiento interno en nuestra plataforma.

Los consumidores Consumer_TOPIC realizan la transformación necesaria y actúan como productores al publicar los mensajes transformados al siguiente exchange.

4. Publicación en Exchange "Direct":

Una vez que los mensajes están en el formato JSON canónico, se publican en el exchange "direct" PurchaseDataJSON utilizando el mecanismo de publicación 'pub'.

El exchange "direct" garantiza que los mensajes ahora en JSON sean enrutados precisamente a la cola que conecta con la pasarela de pago, listos para la transacción.

5. Procesamiento de la Transacción de Pago:

La pasarela de pago, actuando como consumidor, recibe los mensajes de la cola PurchaseDataJSON y procesa las transacciones.

Tras procesar la transacción, la pasarela de pago envía la confirmación de vuelta a través de RabbitMQ, que luego es capturada por la API de Compras para actualizar el estado de la transacción y notificar al usuario.

3. Justificación del diseño

El diseño por el que hemos optado aprovecha la potencia de RabbitMQ para proporcionar un sistema de procesamiento de mensajes altamente confiable y escalable. El uso de exchanges "topic" y "direct" nos permite manejar una variedad de formatos de datos y simplificarlos en un formato estándar para el procesamiento de transacciones.

A continuación presentamos justificaciones adicionales del uso de los diferentes tipos de colas en nuestro diseño:

Diversidad de Datos: El exchange "topic" maneja diferentes formatos de datos de transacciones (JSON, XML, CSV), lo que es crucial para una plataforma como la nuestra, que recibe información de múltiples orígenes o usuarios.

Normalización de Datos: Los consumidores específicos transforman todos los formatos a JSON, creando un estándar o cánón que simplifica el procesamiento de pagos.

Enrutamiento Efectivo: El exchange "direct" envía los datos normalizados directamente a la pasarela de pago, asegurando un proceso de transacción eficiente y sin errores.

Escalabilidad y Mantenimiento: El sistema permite una fácil expansión y un mantenimiento sencillo, vital para adaptarse al crecimiento de la plataforma y a la incorporación de nuevos métodos de pago.

