**Patrón SAGA**

El patrón SAGA, en el contexto de la arquitectura de software, se refiere a un estilo de diseño que se utiliza para manejar transacciones distribuidas. Este patrón se divide en dos tipos principales: el tipo orquestador y el tipo coreografía. La documentación se centra en el tipo de coreografía.

**Patrón SAGA Tipo Coreografía:**

En el patrón SAGA de tipo coreografía, las interacciones entre los servicios se manejan de manera descentralizada, sin un componente central que coordine la ejecución de las transacciones. En lugar de tener un orquestador centralizado, cada servicio participante en la transacción sigue un conjunto de reglas predefinidas para coordinar sus acciones con otros servicios.

**Conceptos clave:**

**Eventos**: Los servicios participantes se comunican entre sí mediante eventos. Un evento es una notificación que indica que algo ha sucedido en un servicio y que otros servicios pueden necesitar conocer.

**Acciones compensatorias**: Cada servicio define acciones que deben llevarse a cabo si se produce un error o una situación no deseada. Estas acciones compensatorias se utilizan para deshacer o revertir las acciones realizadas anteriormente.

**Reglas de coordinación**: Cada servicio sigue un conjunto de reglas de coordinación que determinan cómo debe reaccionar ante ciertos eventos. Estas reglas se definen de antemano y son parte integral de la lógica de negocio del servicio.

**Escenario de Uso:**

Un sistema de comercio electrónico distribuido con varios servicios, como el servicio de inventario, el servicio de pedidos y el servicio de facturación.

**Realización de un pedido:**

Cuando un cliente realiza un pedido, el servicio de pedidos emite un evento de "pedido realizado" que contiene toda la información necesaria.

El servicio de inventario recibe este evento y verifica si hay suficientes productos en stock. Si es así, reduce la cantidad en stock y emite un evento de "inventario actualizado".

Simultáneamente, el servicio de facturación recibe el evento de "pedido realizado" y genera una factura, emitiendo un evento de "factura generada".

Compensación en caso de error:

Supongamos que, después de actualizar el inventario, el servicio de facturación encuentra un problema al generar la factura (por ejemplo, tarjeta de crédito no válida).

El servicio de facturación emite un evento de "error en la facturación", y otros servicios deben reaccionar a este evento.

El servicio de inventario puede recibir el evento de error y revertir la reducción en el inventario mediante un evento de "revertir actualización de inventario".