

Ejercicio: Sistema de notificaciones multi-canal (Teams, Email, Llamadas)



Descripción conceptual: Desarrollaremos un sistema de notificaciones centralizado que:

1. Recibe eventos/alertas de diferentes sistemas (monitoreo, negocio, seguridad)
2. Distribuye estas notificaciones a través de diferentes canales según la prioridad:
 - Teams para la mayoría de notificaciones (alta frecuencia, baja prioridad)
 - Email para notificaciones importantes (frecuencia media, prioridad media)
 - Llamadas telefónicas solo para emergencias críticas (baja frecuencia, alta prioridad)
3. Mantiene un historial de notificaciones recientes
4. Se integra con APIs externas para cada canal de notificación

Tipos de Sinks a utilizar:

1. `Sinks.many().multicast().onBackpressureBuffer()`- Para el flujo principal de eventos/alertas
 - Recibe todos los eventos del sistema
 - Múltiples procesadores se suscribirán a este flujo para enrutar a diferentes canales
 - El buffer es crucial porque no podemos perder notificaciones importantes
2. `Sinks.many().replay().limit(50)`- Para el historial de notificaciones
 - Mantiene un registro de las últimas 50 notificaciones para consulta rápida
 - Útil para paneles de administración o resolución de problemas
 - Permite a nuevos suscriptores ver inmediatamente el contexto reciente
3. `Sinks.one()`(múltiples instancias) - Para las integraciones con APIs externas
 - Una instancia para la API de Teams (relativamente rápida pero no reactiva)
 - Otra para el servicio de email (potencialmente lento)
 - Otra para el sistema de llamadas telefónicas (API externa bloqueante)

Lógica del ejercicio:

1. El sistema recibe constantemente eventos de diferentes fuentes
2. Cada evento lleva una prioridad (BAJA, MEDIA, ALTA) y un mensaje
3. Según la prioridad, el evento se enruta a uno o más canales:
 - BAJA: Solo Teams
 - MEDIA: Teams + Email
 - ALTA: Teams + Email + Llamada telefónica
4. Cada canal tiene su propia lógica de procesamiento y manejo de errores
5. Para cada canal, usamos un Sink.one() con un Scheduler adecuado para manejar la API externa
6. El historial mantiene todas las notificaciones independientemente del canal