

“Documento ITERACION 5”

Esteban Reyes Marcelo, Juan Camilo Sánchez.

Contexto de Presentación del documento

Universidad de los Andes, Bogotá, Colombia

{e.reyesm, jc.sanchez16}@uniandes.edu.co

Fecha de presentación: Diciembre 11 de 2017

Tabla de contenido

1 Análisis de la implementación de transacciones distribuidas; **Error! Marcador no definido.**

2 Especificación e implementación de transacciones distribuidas; **Error! Marcador no definido.**

1 Análisis.

- **Restricciones:**

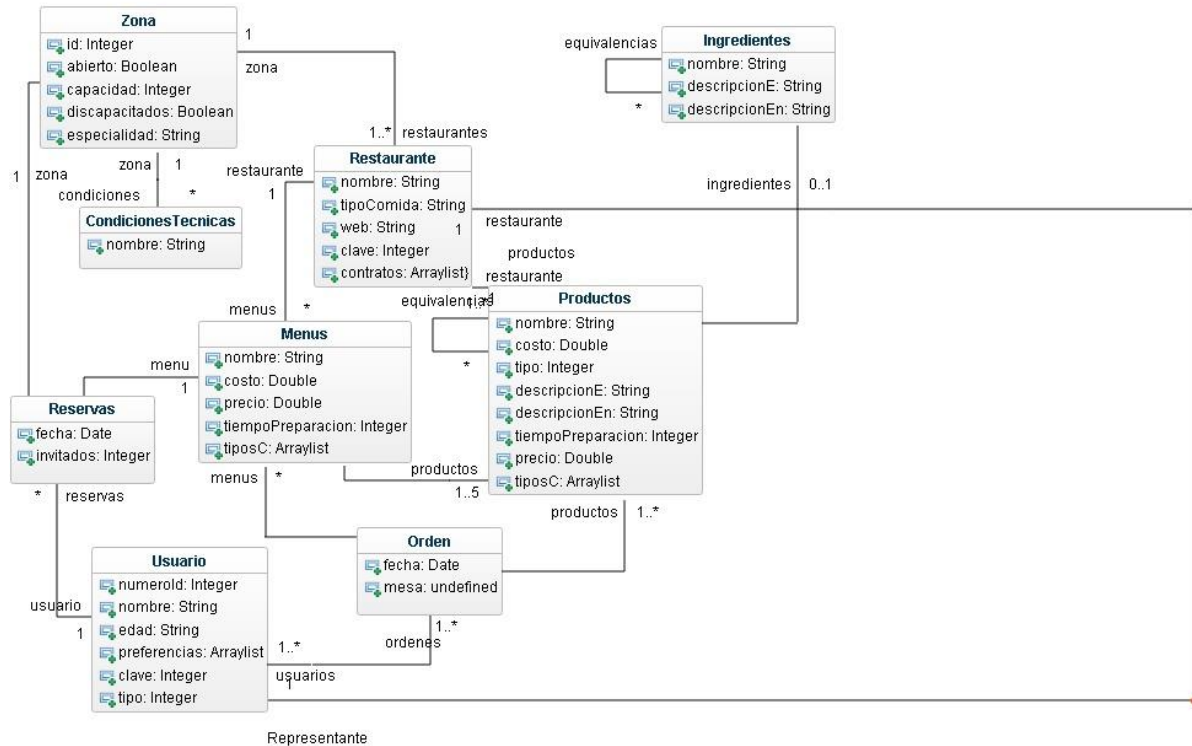
- Algún sitio no cuente con la interfaz XA:

- En el caso que algunas de las rotondas no cuenten con la interfaz XA, esta no podría hacer acciones si se maneja un protocolo TWO PHASE COMMIT, ya que esta no tendría ninguna forma de avisar si está lista y realizar un COMMIT/ROLLBACK posterior a esta acción.

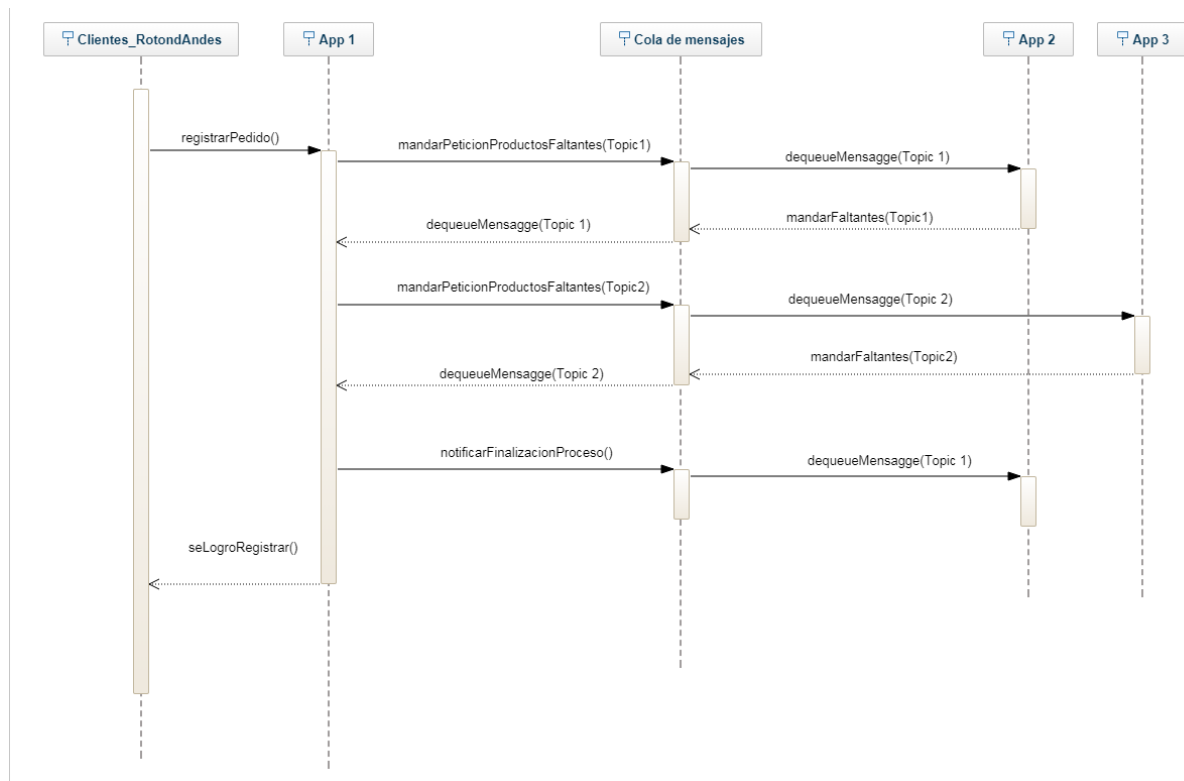
- Alguna rotonda no esté suscrita a un tópico:

- Si algunas de las rotondas no se encuentran suscritas a un tópico, este no tendría una cola en la cual revisar las peticiones del cliente.

- **Diagrama relacional:**



- Diagrama de secuencia para RF18:



En el anterior diagrama muestra como manejaremos el protocolo para el desarrollo del requerimiento funcional 18 (por colas), primero el cliente pide la petición de registrar el pedido a cualquier aplicación (En este caso a la aplicación 1), luego esta aplicación valida cuales son los productos que le faltan para completar el pedido, ya con esto manda a la cola de mensajes

los productos faltantes al topic 1, ya después como la aplicación 2 está pendiente de la cola en el topic 1 recibe los productos que le faltaron a la aplicación 1, valida cuales productos tiene de los faltantes y finalmente responde a la cola en el topic 1 los productos que faltan.

Luego la Cola de mensajes le responde a la aplicación 1 los productos que faltan después que la aplicación 2 ya valido cuales productos pudo proveer; después la aplicación 1 manda a la cola los productos que faltan finalmente por el topic 2, con esto la aplicación 3 quien está pendiente del topic 2 recibe los productos que faltan, analiza cuales él tiene para después responder a la cola los faltantes en total (si la aplicación 3 no tiene todos los productos faltantes realiza roll back).

Finalmente, la cola de mensajes responde a la aplicación 1 los productos que faltan después de haber pasado por todas las aplicaciones, ya con esto la aplicación 1 valida si los productos faltantes son 0 o >0 para realizar roll back o commit y con esto terminar notificándole a la aplicación 2 (si realiza o no roll back) y al cliente como termino su petición.

- **Análisis de las estrategias para desarrollar RF18 por medio de COLAS.**
 - 1) Subdividir el proceso de registrar un pedido en diferentes fases las cuales facilitan la partición de los productos para que cada aplicación logre complementarse y conseguir registrar un pedido correctamente.
 - 2) Asignar tópicos diferentes para la comunicación entre cada par de aplicaciones con el fin de que entre ellas no se repita un mismo producto para registrar.
 - 3) La ultima aplicación es quien decide si realiza roll back o commit notificando a las otras aplicaciones para con esto no violar las reglas de negocio.
 - 4) Manejar turnos para manejar más automático.
- **Análisis de las estrategias para desarrollar RF18 por medio de TWO PHASE COMMIT.**
 - 1) Subdividir el proceso de registrar un pedido en diferentes fases las cuales facilitan la partición de los productos para que cada aplicación logre complementarse y conseguir registrar un pedido correctamente.
 - 2) La aplicación padre es decir la cual le llega la petición de realizar el pedido es quien decide si se realiza un roll back o un commit luego comunicarle a la aplicación faltante en este caso sería la 2 (la aplicación 3 automáticamente sabe que tiene que hacer roll back).
 - 3) Manejar turnos para manejar más automático
- **Comparación de resultados anteriores**

Ambos protocolos cumplen la regla del negocio, debido a la naturaleza del requerimiento la comunicación por colas se

implementará usando algunas ideas del protocolo two phase commit. La única diferencia es que las colas permitirían un comportamiento asíncrono de las aplicaciones.

2 Especificación e implementación.

i. Especificación de la implementación del requerimiento RFC13.

El protocolo a usar es: TWO PHASE COMMIT esto debido a que es un requerimiento diseñado para los clientes y pretendemos una respuesta veloz. Decidimos escoger el método de comunicación síncrono porque precisamente se espera que las demás aplicaciones estén disponibles para otorgar una respuesta inmediata. Además de ser un protocolo que garantiza el cumplimiento de la regla del negocio debido a que no da una respuesta hasta tener los datos de todas las aplicaciones.

ii. Especificación de la implementación del requerimiento RFC14.

Debido a que no es un requerimiento de respuesta rápida decidimos escoger comunicación por COLAS de mensajes debido a que estas nos permiten enviar el mensaje a las demás aplicaciones de manera asíncrona dando la posibilidad de cumplir otros requerimientos con más prioridades mientras se obtiene la respuesta.

iii. Análisis del impacto en las estrategias de transacciones distribuidas.

a. RF18

Analizando el caso de estudio es fundamental que este requerimiento se hubiera implementado por colas, debido a que cabe la posibilidad de que no exista una confirmación de forma inmediata por parte de alguna aplicación y esto no debería frenar el trabajo de la aplicación.

b. RF19 (por colas)

Por colas se puede implementar sin embargo por el comportamiento asíncrono que permite este protocolo de comunicación pueden pasar ciertos problemas que violan las reglas de negocio, esto gracias a que puede presentarse que la aplicación 1 siga corriendo sin que alguna de las dos aplicaciones hubiera eliminado el restaurante lo que permitiría errores en la compra de productos del restaurante el cual se supone se encuentra fuera de RotondAndes.

c. RF19 (por two phase commit)

Por two phase commit nos parece la mejor opción, esto debido a que no permite el funcionamiento de la aplicación hasta que el restaurante se haya eliminado de todas las aplicaciones lo que implica un correcto funcionamiento de la aplicación.