



Catálogo

- Atende as páginas de pesquisa
- Associa destinos aéreos com hotéis (combinações possíveis)
- Faz cache dos resultados vindos dos outros serviços

Reservas

- Atende as páginas de reserva: seleção de destinos, horários, hotéis, pagamento
- Não é sua intenção coordenar os passos da transação, isso é modelado como um SAGAS coreografado pelos próprios serviços de passagens, hotéis e de pagamentos

Um desafio

- Usuário seleciona dia, horário e assento: Usuário > Reservas (interface) > Passagens (backend)
- A resposta demora para chegar e estoura o timeout
- Mas lá no serviço da ponta a coisa prossegue
- Quem capturou o erro tem que enviar a mensagem de cancelamento! Por ex., se Reservas toma um timeout de Passagens.
- Já com o front é mais complicado em caso de perda de conexão, por exemplo. Simplesmente sem resposta, ele não deve continuar, e a requisição que teve sucesso no backend deve expirar e morrer de alguma forma (a menos que tenha sido a finalização, aí só notificar por e-mail, etc.).

- Eu (**Dêrço**) gosto de resolver da seguinte forma: modelo tudo simples, forço o problema e penso o que cada parte pode fazer. Evito antecipação.

Passagens e Hotéis

- Seleção de voo, hotel, quarto, horários: pré-reserva por X minutos
- Mudança de seleção: libera a pré-reserva e realiza outra
- Cancelamento de pré-reserva: acionado em caso de timeout (tempo para realizar toda a transação)...
- Confirmação de uma pré-reserva
- Cancelamento de reserva confirmada: em caso de erro ou timeout em algum serviço (exceto timeout no último, sem que tenha levantado erro)

Mensagens

- Pré-reserva: muda o estado do item (assento no voo/quarto do hotel no dia/horário)
 - Possivelmente cancelando uma pré-reserva anterior feita, em caso de edição
 - A partir da pré-reserva do primeiro item deve ser agendado um evento de timeout
- Confirmação: também muda o estado, agora está efetivamente reservado e o timeout não pode mais ter efeito. Dispara um comando de confirmação para o próximo serviço na cadeia, segundo o SAGAS. Em caso de erro, ao invés, dispara um comando de cancelamento para o item anterior.
 - O item tem que estar no estado de pré-reserva!
- Cancelamento: reverte uma confirmação feita anteriormente e repassa a ordem para trás na cadeia, segundo o SAGAS. Os itens podem continuar em pré-reserva.
- Timeout: cancela uma pré-reserva. Nada deve fazer com uma reserva confirmada (idealmente o timeout deve ser cancelado, mas vai que...!)

Aqui já salta aos olhos a dificuldade de inserir ou remover etapas na cadeia...

Pagamentos

- Interage com os sistemas de pagamento

Mensagens

- Confirmação: dispara uma requisição para um sistema de pagamento externo.

Mensagens externas dos sistemas de pagamento

- Confirmação (de um sistema de pagamento externo): deve disparar as confirmações em cadeia, iniciando o processo SAGAS.

- Erro no pagamento: deve disparar um cancelamento para a etapa anterior, iniciando o processo SAGAS de reversão.