



Ministério da Educação

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO
PARANÁ**

Campus Curitiba

Disciplina: Sistemas Distribuídos

Professora: Ana Cristina Barreiras Kochem Vendramin

Avaliação 1 (valor 2,0)

Aplicação de Leilão

Eventos e Notificações. Microserviços. Chaves assimétricas.

Microserviços (MS) são serviços menores e independentes. Cada microserviço é projetado para ter um conjunto de recursos e é dedicado à solução de um problema específico. Vantagens do uso de Middlewares

Orientados a Mensagens em Microserviços:

- **Desacoplamento:** os microserviços podem operar de forma independente, sem se conhecerem. Cada serviço escuta as filas que interessa e reage de forma autônoma;
- **Escalabilidade:** se houver um aumento na demanda, podemos escalar apenas os microserviços que precisam de mais recursos (por exemplo, aumentar a quantidade de instâncias de um MS);
- **Assincronismo:** os microserviços podem processar suas tarefas em paralelo, sem que um serviço precise esperar por outro para continuar. Isso melhora o desempenho e a experiência do usuário;
- **Flexibilidade:** novos microserviços podem ser facilmente adicionados ao sistema.

Utilize **qualquer linguagem de programação** para desenvolver um sistema de leilão composto por 3 **microserviços**, cada um com uma responsabilidade específica. Os microserviços vão se comunicar exclusivamente via filas de mensagens. O fluxo de dados é orquestrado por eventos através do **serviço de mensageria RabbitMQ** e do **protocolo AMQP** que garantem a sincronização entre as diferentes funcionalidades.

Considere os seguintes processos:

1. **RabbitMQ**

- Middleware orientado a Mensagens (MOM – *Message Oriented Middleware*) responsável por organizar mensagens (eventos) em filas, onde os produtores (*publishers*) as enviam e os consumidores/assinantes (*subscribers*) as recebem.

2. **(0,5) Cliente** (*publisher* e *subscriber*)

- Não se comunica diretamente com nenhum serviço, toda a comunicação é indireta através de filas de mensagens.
 - (0,1) Logo ao inicializar, atuará como consumidor recebendo eventos da fila **leilao_iniciado**. Os eventos recebidos contêm ID do leilão, descrição, data e hora de início e fim.
 - (0,2) Possui um par de chaves pública/privada. Publica lances na fila de mensagens **lance_realizado**. Cada lance contém: ID do leilão, ID do usuário, valor do lance. O cliente assina digitalmente cada lance com sua chave privada.
 - (0,2) Ao dar um lance em um leilão, o cliente atuará como consumidor desse leilão, registrando interesse em receber notificações quando um novo lance for efetuado no leilão de seu interesse ou quando o leilão for encerrado. Por exemplo, se o cliente der um lance no leilão de ID 1, ele escutará a fila **leilao_1**.
3. **(0,3) MS Leilão (publisher)**

- (0,1) Mantém internamente uma lista pré-configurada (*hardcoded*) de leilões com: ID do leilão, descrição, data e hora de início e fim, status (ativo, encerrado).
- (0,1) O leilão de um determinado produto deve ser iniciado quando o tempo definido para esse leilão for atingido. Quando um leilão começa, ele publica o evento na fila: **leilao_iniciado**.
- (0,1) O leilão de um determinado produto deve ser finalizado quando o tempo definido para esse leilão expirar. Quando um leilão termina, ele publica o evento na fila: **leilao_finalizado**.

4. **(0,8) MS Lance (publisher e subscriber)**

- Possui as chaves públicas de todos os clientes.
- (0,2) Escuta os eventos das filas **lance_realizado**, **leilao_iniciado** e **leilao_finalizado**.
- (0,3) Recebe lances de usuários (ID do leilão; ID do usuário, valor do lance) e checka a assinatura digital da mensagem utilizando a chave pública correspondente. Somente aceitará o lance se:
 - A assinatura for válida;
 - ID do leilão existir e se o leilão estiver ativo;
 - Se o lance for maior que o último lance registrado;
- (0,1) Se o lance for válido, o MS Lance publica o evento na fila **lance_validado**.
- (0,2) Ao finalizar um leilão, deve publicar na fila **leilao_vencedor**, informando o ID do leilão, o ID do vencedor do leilão e o valor negociado. O vencedor é o que efetuou o maior lance válido até o encerramento.

5. **(0,4) MS Notificação (publisher e subscriber)**

- (0,2) Escuta os eventos das filas **lance_validado** e **leilao_vencedor**.
- (0,2) Publica esses eventos nas filas específicas para cada leilão, de acordo com o seu ID (**leilao_1**, **leilao_2**, ...), de modo que somente os consumidores interessados nesses leilões recebam as notificações correspondentes.

Observações:

- Desenvolva uma interface com recursos de interação apropriados. •
- **É obrigatória a defesa da aplicação para obter a nota.** •
- O desenvolvimento do sistema pode ser individual ou em dupla.

A Figura 1 ilustra os processos e filas. Na figura consideram-se apenas dois clientes: o ClienteA interessado nos leilões 1 e 2 e o ClienteB interessado apenas no leilão 1.

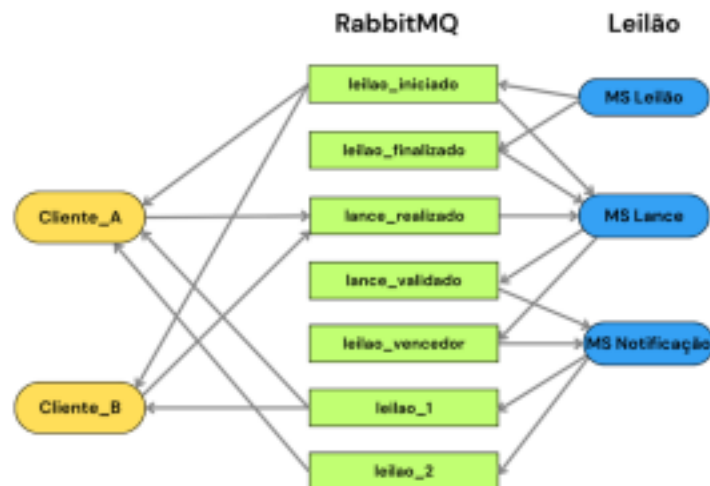


Figura 1. Processos e Filas