



Vamos praticar!

Para a plataforma de educação online, onde os usuários compram um plano e podem acessar diferentes cursos, nos pediram para migrar alguns de seus serviços para Java usando o Spring Cloud.

Para esta etapa, vamos começar com o microserviço encarregado de administrar os diferentes Planos/Subscrições e com o microserviço encarregado de administrar os cursos.

Serviço de Assinatura:

Este serviço tem duas entidades - assinatura e plano - onde a assinatura tem um plano. Detalhes:

- Plan
 - Id
 - name
 - price
- Subscription
 - id
 - startDate
 - endDate
 - plan
 - userId

Nosso tech lead nos pede para:

- Criar um projeto e configurá-lo com o Eureka server.
- Criar o serviço de assinatura e configurá-lo com Eureka Client.
- O nome com o qual você se registra em Eureka deve ser subscription-service.
- Expor uma API que nos permite buscar uma assinatura passando o ID do usuário como parâmetro.
 - O endpoint deve ser /subscription/find e como parâmetro userId. Exemplo: <http://localhost:8081/subscription/find?userId=1>.
 - Implementar JpaRepository com H2 como um banco de dados (opcional).

Serviço do curso:

Este serviço também tem duas entidades - curso e capítulo - onde cada curso tem uma lista de capítulos. Detalhes:

- Course
 - id



- name
 - chapters
- Chapter
 - id
 - name
 - urlStreaming
 - course

Nosso tech lead nos pede para:

- Criar o serviço do curso e configurá-lo como Eureka Client.
- O nome com o qual você se registra em Eureka deve ser course-service.
- Expor uma API que nos permite obter um curso através do ID do curso.
- O endpoint deve ser `/courses/{courseId}`.
 - Além disso, este endpoint recebe a identificação do usuário por header.
- Antes de procurar o curso, validar que o usuário tenha uma assinatura ativa no momento.
 - Usando Feign, procurar a assinatura do usuário (microserviço de assinaturas) e validar que ela está ativa, comparando a data atual com a data de início e fim da assinatura. Se estiver ativo, retorne o curso. Se não estiver ativo, devolva uma mensagem.
- Implementar JpaRepository com H2 como um banco de dados (opcional).