

CHALLENGE - Sistema de críticas de filmes

Descrição

Antes de começar o desafio, é importante informar que esse modelo de desafio é um modelo exaustivo, ou seja, não esperamos que você entregue tudo que está aqui, mas que desenvolva o máximo que conseguir com qualidade. Não se preocupe em entregar tudo, o importante é ir parte por parte e dar o seu melhor!

PARTE 1 - Sistema e regras de negócio

Quando vamos assistir um filme é comum pedirmos indicações. Queremos saber o que as pessoas acharam do filme, se é algo que vale a pena ser assistido. Portanto, crie um sistema que tenha a finalidade de receber e armazenar comentários e notas de um filme.

Crie um api onde um usuário poderá se cadastrar. Cada usuário terá um perfil na plataforma, sendo eles: LEITOR, BÁSICO, AVANÇADO, e MODERADOR. Todo usuário deve começar como LEITOR e poderá ir avançando de perfil conforme a interação com a plataforma.

LEITOR: Após o cadastro, esse usuário poderá logar e buscar por um filme. Ele poderá ver as informações de um filme, comentários e dar uma nota para o filme. A cada filme que o usuário avaliar, ele ganha 1 ponto em seu perfil.

BÁSICO: O usuário leitor poderá se tornar BÁSICO quando adquirir 20 pontos. Nesse perfil será possível postar comentários, notas e responder comentários. Cada resposta que o usuário enviar, ele ganha 1 ponto.

AVANÇADO: O usuário básico poderá se tornar AVANÇADO quando adquirir 100 pontos. Esse perfil tem as capacidades do BÁSICO, e mais citar outros comentários (comentários feitos por outros usuários) e marcar comentários como “gostei” ou “não gostei”.

MODERADOR: Um usuário poderá se tornar MODERADOR de 2 formas: um moderador torna outro usuário moderador ou por pontuação, para se tornar MODERADOR o usuário deverá ter 1000 pontos. O moderador tem as capacidades do AVANÇADO, e mais excluir um comentário ou marcar como repetida.

A busca pelo filme na sua api deve ser feita consultando uma api pública chamada OMDb API (<https://www.omdbapi.com/>) e os comentários e notas devem ser salvos no seu sistema.

PARTE 2 - Segurança

A segurança de um sistema é um dos pontos mais importantes que precisamos ter para garantir uma maior confiabilidade para os nossos clientes e para o nosso próprio projeto. Quando falamos de segurança em um projeto, podemos ir além de funcionalidades básicas que linguagens e frameworks nos entregam. Podemos pensar, arquitetar e desenvolver soluções totalmente voltadas para promover uma maior segurança.

Com base nisso, neste desafio você deve criar uma nova API de autorização de login.

Para que um usuário cadastrado possa realizar as operações no seu sistema, ele deve se autenticar. Com isso, crie uma nova API que ficará somente responsável de autenticar esse usuário.

Quando sua API de críticas receber uma requisição de login com os dados do usuário de login e senha, ela deverá fazer uma requisição para a API de autenticação passando as informações de login e senha.

Sua API de autenticação deverá fazer a validação daquele login e senha estão corretos. Caso esteja, deverá ser gerado um token e devolvido para a API de críticas, que devolverá para o usuário.

Caso o login e senha estiverem errado, sua API de autenticação deverá salvar um cache com uma tentativa de login e a cada nova tentativa de login errada esse cache deve ser atualizado.

Requisitos

- Um usuário não poderá logar sem ter feito um cadastro
- Um usuário não poderá ver filmes e comentários e notas sem estar logado
- Um usuário não poderá criar, editar ou excluir comentários e notas sem estar logado
- Um usuário de um determinado perfil não poderá realizar ações que não fazem parte de seu perfil
- Todas as funcionalidade de seu sistema devem receber um token de autenticação, gerados pela sua API de autorização
- Um usuário não autenticado(que não possui o token) não poderá realizar ações no sistema.
- Um usuário com token invalido não poderá realizar ações no sistema.
- Todas as tentativas falhas de login devem ser salvas em um cache.
- Caso um usuário tente 3 vezes logar e erre, na 4 vez deverá ser retornado uma mensagem de “limite de tentativas excedido “

Requisitos não funcionais

- Armazene os dados de cadastro de usuário e produtos em uma base de dados da sua escolha, inclusive H2.
- Crie um cache para tentativas de login, o cache pode ser utilizando um provedor de cache ou o próprio hashMap do java
- Desenvolva suas API's utilizando a linguagem JAVA.
- Você deve utilizar os filmes fornecidos pela API <https://www.omdbapi.com/>, ou seja deverá consumir essa API na sua aplicação.

Arquitetura da solução

