

Movies Battle

Descrição

Crie uma API REST para uma aplicação ao estilo card game, onde serão informados dois filmes e o jogador deve acertar aquele que possui melhor avaliação no IMDB.

Requisitos

1. O jogador deve fazer login para iniciar uma nova partida. Portanto, cada partida sempre será identificada pela autenticação do usuário.
 - a. Não há restrições onde armazenar os usuários: em memória, ou em banco, etc.
2. Cada rodada do jogo consiste em informar um par de filmes, observando para não repetir o mesmo par nem formar um par com um único filme.
 - a. São sequências não-válidas: [A-A] o mesmo filme repetido; [A-B, A-B] pares repetidos – considere iguais os pares do tipo A-B e B-A.
 - b. Os seguintes pares são válidos: [A-B, B-C] o filme B é usado em pares diferentes.
3. O jogador deve tentar acertar qual filme possui maior pontuação, composta pela nota (0.0-10.0) multiplicado pelo total de votos.
4. Se escolher o vencedor correto, conta 1 ponto. São permitidos até três erros. Após responder, terá acesso a novo par de filmes quando acessar o endpoint do quiz.
5. Forneça endpoints específicos para iniciar e encerrar a partida a qualquer momento. Valide o momento em que cada funcionalidade pode ser acionada.
6. Não deve ser possível avançar para o próximo par sem responder o atual.
7. Deve existir uma funcionalidade de ranking, exibindo os melhores jogadores e suas pontuações.
8. A pontuação é obtida multiplicando a quantidade de quizzes respondidos pela porcentagem de acerto.

Não-Funcionais

1. Armazene os dados em H2 e preencha todas as tabelas necessárias.
2. Inicie os dados de sua aplicação usando webscraping ou a partir da API pública “<http://www.omdbapi.com/>” – levamos a sério que os dados sejam fidedignos.
3. Explore os frameworks Spring: Web, Boot, Data, Security e Cloud.
4. Linguagem: Java 11 ou 17
5. Escreva testes unitários (para validar as regras de negócio) e de integração (para validar a API). Cobertura de testes mínima: 80% dos métodos.
6. Não deixe de adicionar a documentação da API com base no OpenAPI 3.0.
7. Escolha a solução de autenticação que achar mais interessante. Crie pelo menos dois usuários/jogadores.

Teoria

Buscamos os melhores profissionais do mercado que estejam dispostos a fazer parte do nosso time de professores. Por isso, entendemos se você não tiver tempo para codificar a aplicação inteira. Portanto, esta é uma alternativa ao projeto de código, ou seja, **você pode optar por codificar a aplicação OU escrever o seguinte artigo**. Escreva um artigo, de até 2 páginas (no formato: introdução, teoria, desenvolvimento, conclusão), com uma introdução teórica a interfaces, injeção de dependências e Spring Web. Em seguida, escreva uma sessão prática de como as teorias apresentadas podem ser aplicadas para resolver o desafio proposto da aplicação Movies Battle. É fundamental apresentar trechos de código.

Avaliação

Prezamos pelo alinhamento de teoria e prática. Portanto, devem ser aplicadas as boas práticas de desenvolvimento, como código limpo, uso correto de tipos, operadores e bibliotecas. Por isso, busque escrever o código o mais enxuto possível; somos do time “menos é mais”.