

Case de Seleção — Backend Developer

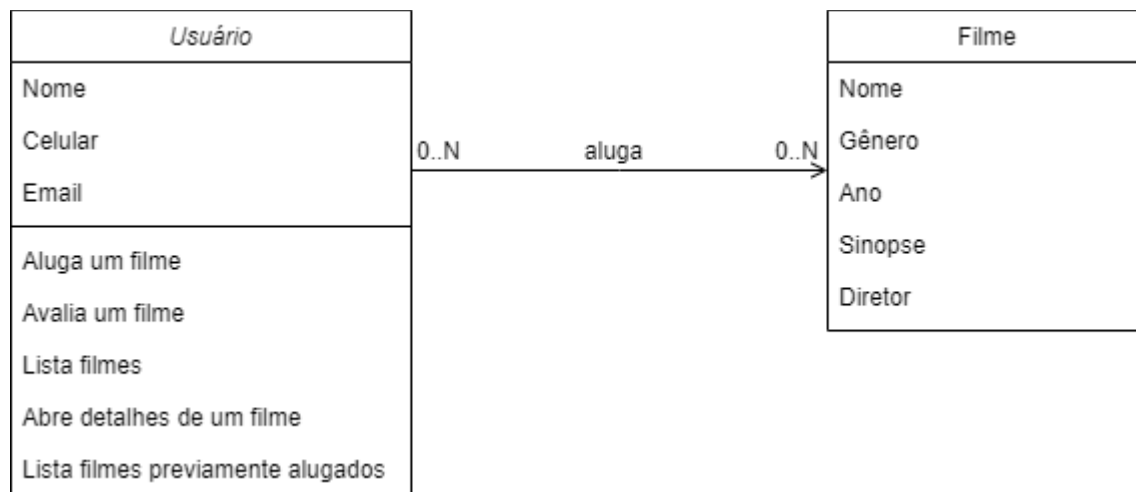
Você foi contratado pela FilmesTop.com para desenvolver uma API para que os seus usuários possam navegar pelo site da empresa.

A empresa determinou que esta API deve ser desenvolvida em Python utilizando o padrão REST, e que o banco de dados deve ser relacional.

As funcionalidades do usuário que estão sob sua responsabilidade são:

1. o usuário deve ser capaz de visualizar a lista de filmes disponíveis por gênero;
2. o usuário deve ser capaz de listar todas as informações sobre um determinado filme;
3. o usuário deve ser capaz de alugar um filme;
4. o usuário deve ser capaz de associar uma nota a cada filme já alugado;
5. o usuário deve ser capaz de visualizar todos os filmes que ele já alugou com as notas que ele atribuiu para cada filme e a data de locação.

Segue abaixo um diagrama de classes com os atributos de cada classe e os métodos associados a elas:



É necessário que:

1. o código-fonte esteja organizado da melhor forma possível para que sua manutenção ocorra sem maiores problemas. A FilmesTop.com preza por uma ótima qualidade de código;

2. O código fonte esteja disponível em um repositório público como Github ou Gitlab para revisão por seus pares na FilmesTop.com.
3. As funções auxiliares e os endpoints estejam bem documentados (tenha em mente que alguém de front-end precisará consultar a documentação da sua API para utilizá-la corretamente, portanto, forneça uma documentação dos parâmetros de entrada e o formato da saída).

Features opcionais:

- a. Utilizar PostgreSQL como banco de dados ;
- b. Realizar testes do que foi desenvolvido. Tente criar testes úteis para testar diferentes cenários de cada feature desenvolvida;
- c. Permitir que a aplicação seja disponibilizada usando Docker;
- d. Adoção de padrões de arquitetura de código (Por exemplo: singleton, repository, factory, etc);
- e. Adoção de técnicas para melhorar a escalabilidade:
 - i. Cache para consultas recorrentes
 - ii. Threads para processamentos pesados
- f. Versionar as alterações no banco de dados através de migrações;
- g. Adicionar as colunas “nota_final” e “total_avaliacoes”, utilizando uma migração para incorporar essas duas colunas à tabela Filme; e
- h. Adaptar os endpoints para que a cada avaliação sejam incrementadas as colunas “total_avaliacoes” e “nota_final” daquele filme. **Obs.: a coluna nota_final é a média de todas as notas anteriores.**

Tecnologias que devem ser utilizadas:

- Python
- Flask

Entrega Final:

- O candidato deve enviar o link de acesso público ao repositório público como Github ou Gitlab, seguindo os critérios acima até o prazo informado no e-mail
- O repositório deve conter um Readme com as orientações de como executar o projeto
- O candidato deve fazer uma apresentação de **no máximo 7 minutos** sobre o projeto no dia e no horário de sua entrevista

