

Atividade e Objetivo

Tema: Gestão de Biblioteca com Recomendação de Livros

Objetivo: Desenvolver uma aplicação de gestão de biblioteca que permita o cadastro de livros, usuários, empréstimos, devoluções e forneça recomendações de livros. A aplicação deve utilizar um banco de dados relacional com relacionamento entre tabelas e uma API REST para as operações CRUD. Há duas questões bônus: uma para integração com a API do Google Books e outra para o desenvolvimento de um frontend em React ou Angular.

Requisitos da Prova:

1. Modelagem de Banco de Dados

- **Tabela Usuários:**

- Campos: `id`, `nome`, `email`, `data_cadastro`, `telefone`
- Todos os campos não podem ser nulos.
- Validar se email está correto.
- A data de cadastro não pode ser maior que o dia atual.

- **Tabela Livros:**

- Campos: `id`, `titulo`, `autor`, `isbn`, `data_publicacao`, `categoria`
- Todos os campos não podem ser nulos.

- **Tabela Empréstimos:**

- Campos: `id`, `usuario_id`, `livro_id`, `data_emprestimo`, `data_devolucao`, `status`
- Todos os campos não podem ser nulos.
- Data de empréstimo não pode ser maior que o dia atual.

Defina os relacionamentos:

- Um usuário pode fazer vários empréstimos.
- Um livro pode ser emprestado várias vezes, mas apenas um empréstimo ativo por vez.

2. Operações CRUD com API REST

- **Usuários:**
 - Criar, ler, atualizar e deletar usuários.
- **Livros:**
 - Criar, ler, atualizar e deletar livros.
- **Empréstimos:**
 - Criar empréstimos (associar um livro a um usuário).
 - Atualizar empréstimos (alterar status e data de devolução).

3. Recomendação de Livros

- Implemente um sistema de recomendação de livros baseado na categoria dos livros emprestados pelo usuário. Se um usuário já emprestou livros de uma determinada categoria, recomende outros livros da mesma categoria que ele ainda não emprestou.

4. Testes Unitários

- Escreva testes unitários para os serviços principais da aplicação.
- Utilize JUnit e Mockito para a implementação dos testes.

Questões Bônus

Questão Bônus 1: Integração com a API do Google Books

- Implemente uma funcionalidade que permita buscar informações de um livro utilizando a API do Google Books.
- Adicione um endpoint na API REST para buscar livros por título na API do Google Books.
- Permita que o usuário possa adicionar um livro retornado pela busca diretamente na biblioteca.

Questão Bônus 2: Desenvolvimento do Frontend em React ou Angular

- Desenvolva uma interface de usuário para a aplicação utilizando React ou Angular.
- A interface deve permitir a interação com as funcionalidades principais da API REST: cadastro de usuários, cadastro de livros, gerenciamento de empréstimos e visualização de recomendações de livros.

- Adicione uma funcionalidade de busca de livros pela API do Google Books na interface e permita que os livros encontrados sejam adicionados à biblioteca diretamente pelo frontend.

Entrega:

- O código deve estar em um repositório Git (como GitHub ou GitLab).
- Inclua um README com instruções claras sobre como executar a aplicação, incluindo a configuração do banco de dados.
- O prazo para entrega estará descrito no e-mail.

Critérios de Avaliação:

- **Funcionalidade:** A aplicação atende a todos os requisitos especificados.
- **Qualidade do Código:** O código é limpo, bem estruturado e segue as boas práticas de programação.
- **Documentação:** A API está bem documentada e as instruções de execução são claras.
- **Testes:** A presença e a cobertura dos testes unitários.
- **Questões Bônus:** Implementação correta e funcional da integração com a API do Google Books (Não incluir a API KEY no código entregue) e do frontend em React ou Angular.

Boa sorte! 😊