

Oportunidade: Dev. Full-stack Sênior

Desafio Técnico

- Sistema para gerenciar os dados do portfólio de projetos de uma empresa. Entenda como portfólio de projetos o conjunto de projetos da empresa, tanto em andamento como em análise de viabilidade.

Regras de negócio:

- [] O sistema deve permitir o cadastro (inserção, exclusão, alteração e consulta) de projetos. Para cada projeto devem ser informados: nome, data de início, gerente responsável, previsão de término, data real de término, orçamento total, descrição e status.

- [] Os projetos devem ser classificados em: baixo risco, médio risco e alto risco. A classificação de risco não é cadastrada no sistema.

- [] A cada instante, o projeto deve estar em um status específico e único. Os status possíveis não são cadastrados no sistema e são: em análise, análise realizada, análise aprovada, iniciado, planejado, em andamento, encerrado, cancelado.

- [] Se um projeto foi mudado o status para iniciado, em andamento ou encerrado não pode mais ser excluído

- [] O sistema não deve permitir o cadastro de um novo membro diretamente. Deve ser provida funcionalidade via web service, contendo nome e atribuição (cargo).

- [] O sistema deve permitir associar membros aos projetos que tem atribuição funcionário .

Regras de implementação:

- [] • Persistir as informações no BD enviado, utilizando os frameworks solicitados

- [] • Utilizar MVC

- [] • Utilizar testes unitários para avaliar as regras;

- [] • Utilizar testes unitários para avaliar as regras;

- [] • Utilizar JSP com padrão Bootstrap para o front-end

Entrega:

- O teste deverá ser enviado em um e-mail contendo o link para o repositório deste projeto em sua conta pessoal no GitHub.

- Não deverá constar em nenhum lugar do projeto (estruturas de pastas, nomes de arquivos, código fonte, etc) a palavra "Desbravador".

Orientações técnicas:

<https://github.com/DesbravadorSoftware/desafioJava#readme>

Critérios de avaliação:

- [] • Persistir as informações no BD enviado, utilizando os frameworks solicitados (1,0)
- [] • Utilizar MVC (2,0)
- [] • Utilização correta dos conceitos de Orientação a Objetos (2,5)
- [] • Padrão na construção das classes e nas nomenclaturas (1,0)
- [] • Criação de testes unitários. (2,0)
- [] • Implementação das regras de negócio (1,5)