

Papel do Desenvolvedor Backend

- Implementar o design arquitetural em código limpo e de alta qualidade.
- Dominar e aplicar Design Patterns, SOLID e Clean Code, garantindo a manutenibilidade e aderência à arquitetura.
- Dominar o stack técnico (C#, .NET, Docker, IaC) e utilizá-lo de forma eficiente.
- Implementar padrões de integração (APIs, Mensageria) e componentes, respeitando as abstrações (DDD) e o design.
- Aplicar rigorosamente mecanismos de segurança (OIDC/OAuth2 com o SSO) e boas práticas de transporte de dados no código.
- Garantir o atendimento aos requisitos não-funcionais (NFRs) através de código testável (TDD), otimizado e resiliente.

Objetivo do Desafio

Implementar uma solução de software escalável e resiliente, baseada na melhor arquitetura para garantir alta disponibilidade, segurança e desempenho. Isso inclui a aplicação prática de padrões de design, princípios SOLID, e a implementação correta dos requisitos não-funcionais.

Deve abranger aspectos importantes, como design, implementação, testes e segurança.

- **Escalabilidade:** Garanta que a solução implementada possa lidar com o aumento da carga de trabalho. A implementação deve ser stateless e permitir dimensionamento horizontal.
- **Resiliência:** Implemente a solução visando a recuperação de falhas.
- **Segurança:** Proteja os dados e os endpoints. Implemente o básico de autenticação/autorização e trate dados sensíveis.
- **Aplicação de Padrões:** O desafio deve ser implementado usando uma arquitetura de microsserviços e aplicar padrões de design (Design Patterns), princípios SOLID e conceitos de DDD no código.
- **Integração:** Implemente a comunicação entre os componentes.
- **Requisitos Não-Funcionais:** Garanta que a implementação atenda aos requisitos de desempenho e disponibilidade definidos.
- **Documentação:** Registre as decisões de implementação, diagramas (simples) e fluxos de dados. Isso facilita a comunicação e a manutenção.

O foco principal deste desafio é a qualidade da codificação. Embora o desenho da solução seja obrigatório (diagrama), a avaliação principal será sobre a implementação prática (código) que demonstra seu conhecimento empírico, aplicação de boas práticas (SOLID, TDD, DDD), e a qualidade do código para atender os requisitos.

Descriptivo da Solução

Um comerciante precisa controlar o seu fluxo de caixa diário com os lançamentos (débitos e créditos), também precisa de um relatório que disponibilize o saldo diário consolidado.

Requisitos de negócio

- Serviço que faça o controle de lançamentos (Transacional).
- Serviço do consolidado diário (Relatório).

Requisitos técnicos obrigatórios

- Desenho da solução (um diagrama simples de componentes e interações).
- Deve ser feito usando C#.
- Testes (TDD/BDD). O código deve ser testável e os testes devem cobrir a lógica de negócios.
- Boas práticas (SOLID, Clean Code, Design Patterns, DDD).
- Readme com instruções claras de como a aplicação funciona, e como rodar localmente.
- Códigos que não funcionarem localmente seguindo as instruções, podem ser descartados
- Hospedar em repositório público (GitHub).
- Todas as documentações de projeto devem estar no repositório.

Caso os requisitos técnicos obrigatórios não sejam minimamente atendidos, o teste será descartado.

Requisitos não funcionais

- O serviço de controle de lançamento não deve ficar indisponível se o sistema de consolidado diário cair.
- Em dias de picos, o serviço de consolidado diário recebe 50 requisições por segundo, com no máximo 5% de perda de requisições.

Observações

Leve em consideração todos os critérios técnicos mencionados, mas não se prenda somente a eles. Use o teste para demonstrar sua habilidade em tomar decisões sobre o que é importante durante o desenvolvimento de aplicações.

Também são bem vindas descrições sobre o que você gostaria de ter implementado ou evoluções futuras para o sistema proposto. Entendemos que o tempo para a execução do projeto é limitado, então aproveite a documentação do seu

projeto para apresentar pontos que possam demonstrar melhor suas capacidades, além do que é possível demonstrar através do desenvolvimento desse desafio.

Desejamos boa sorte para você!