

# Desafio Java Inter (Cross Border)

## Instruções:

- O Desafio deverá ser desenvolvido de forma incremental, por exemplo, deixe tratamento de cotação nos finais de semana por último.
- O código deverá ser publicado em um repositório privado no GitHub e deverá ser fornecido acesso de leitura para os usuários [hitallo.silva@inter.co](mailto:hitallo.silva@inter.co) e [lucas.rcastro@inter.co](mailto:lucas.rcastro@inter.co).
- Envie o desafio mesmo se não tiver finalizado completamente.
- O Código deverá estar compilando e funcionando
- Desenvolver aplicação utilizando Java e os frameworks Spring ou Micronaut
- Utilizar Maven
- Você deverá criar um arquivo README.md na raiz do seu repositório com as seguintes instruções:
  1. Como compilar e executar a sua aplicação.
  2. Como executar os testes unitários.
  3. Sinta-se livre para adicionar qualquer comentário.

# Objetivo: Serviço de Remessa

Temos 2 tipos de usuários: Pessoa Física (PF) e Pessoa Jurídica (PJ). Ambos possuem carteira em Real e em Dólar e realizam remessa entre eles.

## Requisitos:

- Deverá ser criado uma API RESTFul que fará uma remessa entre os usuários. Uma remessa constitui de uma operação de conversão de moeda de Real para Dólar e depois de uma operação de transferência do valor convertido.
- A cotação da moeda deverá ser consultada através da API publica <https://dadosabertos.bcb.gov.br/dataset/dolar-americano-usd-todos-os-boletins-diarios/resource/22ab054c-b3ff-4864-82f7-b2815c7a77ec>.  
Obs.: utilizar o campo "cotacaoCompra" da resposta da API.
- Utilizar um banco em memória de sua preferência.
- Os usuários deverão ter uma carteira de saldo em Real e saldo em Dólar.
- Validar se o usuário tem saldo antes da remessa.
- Os usuários deverão ser criados com nome completo, e-mail, senha, CPF para PF e CNPJ para PJ.
- Os e-mails, CPF e CNPJ's deverão ser únicos.
- PF's possuem um limite de 10 mil reais transacionados por dia.
- PJ's possuem um limite de 50 mil reais transacionados por dia.
- Não há restrição de remessa entre PF's e PJ's e vice-versa.

- A cotação de moeda não funciona aos finais de semana, sendo assim, o serviço deverá consultar a última cotação de moeda obtida.
- A operação de transferência deve ser uma transação (ou seja, revertida em qualquer caso de inconsistência) e o dinheiro deve voltar para a carteira do usuário que envia.

## Avaliação:

- Documentação
- Testes unitários
- Aplicação de padrões de Design Patterns, SOLID, etc..
- Código limpo e organização do projeto
- Modelagem do banco de dados
- Tratamento de erros
- Acoplamento entre componentes
- Testabilidade e Manutenibilidade do código
- Commits pequenos e descritivos

## Diferencial

- Testes de integração
- Cache
- Melhorias
- Docker e Kubernetes