



## HEY, PRONTOS PARA O DESAFIO FINAL?

O banco fictício **BlueBank** está construindo uma nova plataforma e precisa de uma API para gerenciar as transações. Utilize suas novas habilidades com Java, Banco de dados e AWS.

O sistema deve permitir cadastro de novos clientes, incluindo dados pessoais e dados para contato. O cliente deve ser atrelado a uma conta bancária e registrar histórico de transações entre as contas.

## SUGESTÃO DE ORGANIZAÇÃO DO PROJETO

Elaboração do kanban com definição dos entregáveis

1. Elaboração do kanban (*sugestão de utilização: Trello, Jira, etc*)
2. Criação do backlog (*com tarefas referente ao desenvolvimento*)
3. Detalhamento descritivo das tarefas da squad dentro dos seus cards (e não apenas com títulos genéricos no card)-Nome para a squad
4. Formatação do kanban padrão "to do, doing, done"
5. Definição de data de entrega das tarefas nos cards
6. Definição de responsável pelo card ou checklist de completude
7. Priorização dos cards (ex: tags com cores para maior relevância ou com títulos descritivos para nível de importância na priorização)

## ENTREGAS MÍNIMAS

1. Metodologias Ágeis:
  - a. Kanban com todas as tarefas organizadas e responsáveis definidos.
2. Back-end:
  - a. Cadastro de clientes
  - b. Listagem de clientes
  - c. Atualização de clientes
  - d. Deletar clientes
  - e. Histórico de transações entre contas



3. Banco de Dados:
  - a. Tabelas bem estruturadas e populadas com valores para testes.
  - b. O Banco deve ser entregue em script SQL junto ao repositório.
4. Gerais:
  - a. O código deve ser entregue em um repositório no Github.
  - b. A API deve ser disponibilizada em ambiente AWS com EC2 e em Beanstalk.
  - c. A aplicação deve ter um pipeline em Jenkins ou no Aws Build.
  - d. A aplicação precisa ser configurada no API Gateway da AWS.
  - e. A aplicação precisa ter no mínimo um endpoint de SNS para cadastro de emails e verificação automática.
  - f. A aplicação precisa ter no mínimo um Lambda.
  - g. Liste os endpoints no README.md

## ENTREGA OPCIONAL

1. Testes automatizados, podem ser testes unitários ou testes de integração.

## CRITÉRIOS DE AVALIAÇÃO

1. Documentação da Aplicação;
2. Sugestão de documentação com swagger;