



Caro Asp a Dev, desejamos boas vindas!

Chegou o grande momento de demonstrar os conhecimentos adquiridos até o presente momento deste curso. E por isso, gostaríamos de propor um desafio muito interessante e que poderá ser usado como fonte de amostra em seu portfólio de aplicações.

### Regras Iniciais do Desafio:

- O desafio terá de 3 (três) a 4 (quatro) integrantes, dependendo da disponibilidade.
- O projeto será dividido em 4 (quatro) momentos:
  - 1 Etapa** - Desenvolvimento do protótipo no figma e modelagem do banco de dados.
  - 2 Etapa** - Desenvolvimento da API em Node, com seus respectivos endpoints.
  - 3 Etapa** - Configuração do servidor e criação do banco de dados.
  - 4 Etapa** - Desenvolvimento do front-end usando ReactJS e seus conceitos.
- As etapas deverão ser seguidas de acordo com o cronograma abaixo:

Etapas	Data Início	Data Fim
1	27/05	01/06
2	02/06	07/06
3	08/06	09/06
4	10/06	15/06

- Entre as datas acima descritas, os grupos deverão se comprometer a realizar pelo menos um agendamento com os instrutores, para avaliarem o andamento do desafio. *Datas a definir.*
- Todos os integrantes do grupo deverão participar de forma ativa em pelo duas etapas.

### O desafio:

Um novo banco de um conglomerado inglês está chegando ao Brasil (**Alpha Bank**) . Após meses de negociações para entrada no mercado brasileiro e o desenvolvimento da estratégia do negócio, é chegada a hora do início do desenvolvimento do **internet banking** da instituição. Para tanto, você e mais alguns colegas foram contratados a fim de embarcar nesse desafio e criar uma aplicação do zero que fizesse sentido aos planos do banco. Os executivos do banco estimam que o lançamento do MVP (Minimum Viable Product) da aplicação deva ocorrer nas próximas semanas e para tanto, foi solicitado que a equipe desenvolvesse a aplicação nos seguintes moldes:



### 2º ETAPA:

#### Para equipe de back-end - Construção da API:

- 1º Usar formato JSON para leitura e escrita.
- 2º Criar os endpoints com suas respectivas sub-rotas.
- 3º Validar todos os campos de entrada.
- 4º Fazer o uso de **controllers** e **middlewares**.
- 5º No controle de sessão usar o **JWT**.
- 6º Usar o **uuid** para geração do **id\_cliente** aleatório.
- 7º Usar o **Router** do Express.
- 8º Usar variáveis de ambiente com o **dot-env**.
- 9º Criação, Deleção e Atualização devem possuir **datas** para suas respectivas ações.