



**there**  
by e.Sales

**.desafio técnico**

*/automation test developer*



# .introdução

Bem-vindo ao nosso desafio técnico!

Este exercício foi elaborado para avaliar suas habilidades e conhecimentos na criação e execução de testes automatizados com *Cypress*.

Esperamos que, com este desafio, você possa demonstrar um pouco da sua experiência, capacidade analítica e de resolução de problemas.

Daremos algumas sugestões de como atingir os objetivos do desafio, mas sinta-se livre para utilizar outros padrões, ferramentas ou até mesmo trazer diferentes insights em sua apresentação, caso achar necessário.



# .API de cobrança Pix

## Cenário hipotético

- O time de tecnologia da Theke precisa desenvolver uma integração via API com o *Banco XYZ*, um novo banco digital que surgiu em 2025;
- O *Banco XYZ* nos disponibilizou a documentação *Swagger* completa de uma API para registrar cobranças Pix com vencimento;
- Em reuniões de discovery e refinamento com o time técnico da Theke, foram discutidos inúmeros cenários de requisição e tratamento de respostas dessa API, já que o banco segue à risca o *Schema Json* e as regras expostas na documentação;
- Todavia, o método de autenticação da API se dará por meio de uma *API Key* de nome “X-API-Key”, contendo uma chave alfanumérica de 36 caracteres.

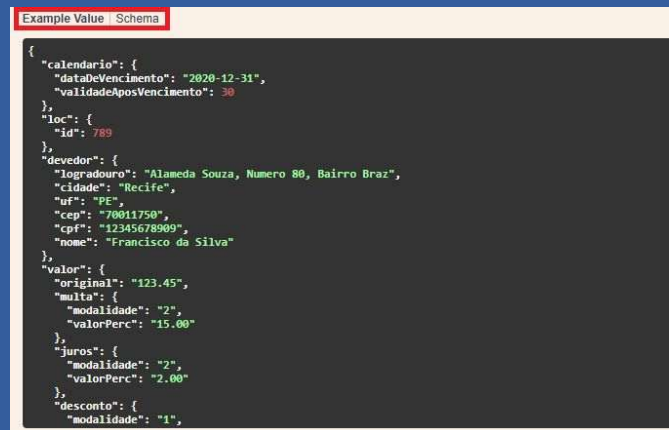
# .API de cobrança Pix

- **Funcionalidade**

Registrar cobranças Pix com vencimento no *Banco XYZ*, para que os clientes finais possam efetuar o pagamento das cobranças registradas via *Pix Copia E Cola* ao invés de boleto bancário com código de barras;

- **Documentação Swagger - Bacen**

[https://bacen.github.io/pix-api/index.html#/CobV/put\\_cobv\\_txid](https://bacen.github.io/pix-api/index.html#/CobV/put_cobv_txid)



```
{
  "calendario": {
    "dataDeVencimento": "2020-12-31",
    "validadeAposVencimento": 30
  },
  "loc": {
    "id": 789
  },
  "devedor": {
    "logradouro": "Alameda Souza, Numero 80, Bairro Braz",
    "cidade": "Recife",
    "uf": "PE",
    "cep": "70011750",
    "cpf": "12345678909",
    "nome": "Francisco da Silva"
  },
  "valor": {
    "original": "123.45",
    "multa": {
      "modalidade": "2",
      "valorPerc": "15.00"
    },
    "juros": {
      "modalidade": "2",
      "valorPerc": "2.00"
    },
    "desconto": {
      "modalidade": "1",
```



# .API de cobrança Pix

## Objetivos

1. Mapear quatro casos de teste (dois positivos e dois negativos) que você considera essenciais no processo de registrar cobranças Pix com vencimento:
  - Documentar os casos de teste de forma detalhada e estruturada;
  - Incluir, ao menos, pré-condições, passos e resultados esperados.
2. Automatizar dois dos quatro cenários, utilizando *Cypress* e um *Mock Server* (para simular a API bancária). **Sugestões:**
  - Utilizar *Page Object Model* para organizar os testes;
  - Garantir a reutilização de componentes;
  - Configurar variáveis de ambiente para armazenar dados sensíveis;
  - *JSON Server*, *WireMock* ou *MSW*;
  - Repositório/projeto com instruções claras para execução;
  - Código funcional, limpo e organizado.
3. Nos apresentar os casos mapeados e o código desenvolvido no próximo encontro.



# /obrigado e boa sorte!

**Pablo Hadler**

[pablo.gomez@esales.com.br](mailto:pablo.gomez@esales.com.br)

**Pedro Cardoso**

[pedro.cardoso@esales.com.br](mailto:pedro.cardoso@esales.com.br)