

# Prova de avaliação

Last updated by | joaopmaa | 27 de jun. de 2024 at 09:18 BRT

---

## Que bom ter você aqui 😊

Olá! Ficamos felizes pelo seu interesse na vaga para dev frontend aqui no [PB](#).

O objetivo deste desafio é avaliar suas habilidades técnicas como pessoa candidata à vaga de frontend React, bem como sua capacidade de resolver problemas de forma eficiente e clara. Lembrando que esse desafio deve ser feito de forma individual.

## O desafio

Implementado em 2020, o saque-aniversário foi uma medida inovadora que permite que o trabalhador tenha acesso parcial ao seu saldo do Fundo de Garantia do Tempo de Serviço (FGTS) enquanto ainda estiver empregado.

Nossos usuários que possuem saldo no FGTS optaram por sacar uma parcela do saldo disponível em sua conta e desejam consultar quanto conseguirão receber no mês do seu aniversário.

Sabendo que o saque-aniversário tem regras específicas para o cálculo do valor a ser sacado, e que esse valor pode variar de acordo com o saldo disponível na conta do trabalhador e outros fatores como a distribuição dos lucros do FGTS, você será responsável por desenvolver uma aplicação web responsiva para atender à essa demanda.

## Requisitos funcionais

- ☐ O website deve ter obrigatoriamente duas páginas com rotas diferentes: uma página inicial e uma página de resultado.
- ☐ A página inicial deve ter um formulário em que o usuário possa preencher o seu nome, telefone completo, saldo do FGTS e o seu mês de aniversário. E também um botão ver proposta.
- ☐ Deve-se validar o telefone preenchido com a [Phone Number Validation and Verification API](#) via fetch, e informar ao usuário caso o número seja inválido;
- ☐ Deve-se utilizar componente html seletor preenchido com os 12 meses do ano, no campo de seleção do mês de aniversário;
- ☐ O usuário deve ter acesso a uma outra página com o resultado com o valor do saque-aniversário do FGTS, com base nas informações fornecidas pelo usuário. Esse resultado deve seguir as seguintes regras:
  - Se o saldo do FGTS na data de aniversário do trabalhador for até R\$ 500, o trabalhador pode sacar 50% do saldo;
  - Se o saldo do FGTS na data de aniversário do trabalhador estiver entre R\$ 500,01 e R\$ 1.000, o trabalhador pode sacar 40% do saldo mais um valor fixo de R\$ 50;
  - Se o saldo do FGTS na data de aniversário do trabalhador estiver entre R\$ 1.000,01 e R\$ 5.000, o trabalhador pode sacar 30% do saldo mais um valor fixo de R\$ 150;
  - Se o saldo do FGTS na data de aniversário do trabalhador estiver entre R\$ 5.000,01 e R\$ 10.000, o trabalhador pode sacar 20% do saldo mais um valor fixo de R\$ 650;
  - Se o saldo do FGTS na data de aniversário do trabalhador estiver entre R\$ 10.000,01 e R\$ 15.000, o trabalhador pode sacar 15% do saldo mais um valor fixo de R\$ 1.150;
  - Se o saldo do FGTS na data de aniversário do trabalhador estiver entre R\$ 15.000,01 e R\$ 20.000, o trabalhador pode sacar 10% do saldo mais um valor fixo de R\$ 1.900;

- Se o saldo do FGTS na data de aniversário do trabalhador for superior a R\$ 20.000,01 o trabalhador pode sacar 5% do saldo mais um valor fixo de R\$ 2.900.

**Exemplo:** O trabalhador que tem R\$ 1 mil no FGTS pode receber de Saque-Aniversário R\$ 400,00 (alíquota de 40%) acrescido de R\$ 50,00 (parcela adicional), totalizando R\$ 450,00.

- ☐ Junto com o resultado, deve-se exibir o nome do usuário preenchido na página anterior;
- ☐ Deve-se seguir o layout em anexo sugerido. Não há necessidade de 100% fiel em pixel, mas respeite a composição, fontes (Montserrat) e cores.
- ☐ Testes de unidade (aqui no PB usamos a testing-library e o Jest). Sabemos que fazer testes leva tempo, mas tente mostrar para nós pelo menos como você faria um setup básico para testes no projeto e alguns cenários que poderiam ser testados;

## Requisitos não funcionais

- ☐ O app deverá ser criado usando [React](#)
- ☐ Na raiz do projeto, inclua um README.md com as instruções para construir seu projeto localmente e com introdução sobre o que é o projeto feito.
- ☐ O app deverá se comportar da mesma forma na última versão estável dos seguintes browsers: Chrome, Firefox, Edge.
- ☐ O website deverá ser responsivo (Desktop e Mobile).
- ☐ O website deve ser disponibilizado no GitHub como um repositório público do próprio candidato.
- ☐ Para conexão com a [Phone Number Validation and Verification API](#) deve-se criar uma conta gratuita e gerar o API TOKEN neste [link](#)

## Será considerado um PLUS+

- ☐ Uso do framework NextJS;
- ☐ Otimizações para performance da aplicação;
- ☐ Trabalhar com estados globais na aplicação e uso de recursos nativos do React (exemplo: usar ContextAPI ao invés de Redux);

## O que vamos avaliar

- Conhecimento em HTML Semântico, criação de componentes funcionais em react, uso de hooks, routes e gerenciamento de estado;
- Habilidades para escrever um código limpo, organizado e de fácil manutenção, seguindo as boas práticas de desenvolvimento de software;
- Habilidade de resolver problemas lógicos;
- Mensagens e mudanças nos commits;
- Habilidade de documentação, sendo capaz de explicar como construir o app localmente, decisões e etc.

---

Muito obrigado e um bom desafio! Dúvidas estamos a disposição.

---

## Arquivo a ser anexado:

[DesafioTecnicoFrontendPB.zip](#)