

# **Bootcamp: Desenvolvedor Full Stack**

#### Desafio

Módulo 1	Fundamentos	
$\times$		

# **Objetivos**

Exercitar os seguintes conceitos trabalhados no Módulo:

- ✓ Declarar elementos HTML como títulos, input, div, span, ul, li etc.
- ✓ Estilizar o app com CSS.
- ✓ Mapear elementos do DOM para serem manipulados com JavaScript.
- ✓ Formatar valores com JavaScript.
- ✓ Realizar diversos cálculos com array methods como map, filter e reduce.
- ✓ Realizar requisições HTTP com o comando fetch.

#### Enunciado

Criar uma aplicação para filtrar usuários e mostrar estatísticas a partir do filtro definido.

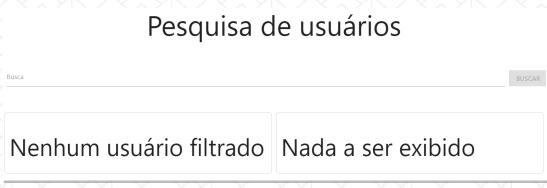
#### **Atividades**

Os alunos deverão desempenhar as seguintes atividades:

- Na carga inicial da aplicação, obter os dados de:
  <a href="https://randomuser.me/api/?seed=javascript&results=100&nat=BR&noinfo">https://randomuser.me/api/?seed=javascript&results=100&nat=BR&noinfo</a>
- 2. Carregar os dados dos usuários em um array.

- 3. Permitir a filtragem de usuários através de um input com interação do usuário.
- 4. O usuário poderá filtrar dados quando digitar pelo menos um caractere no input.
- 5. O usuário poderá filtrar os dados tanto digitando "Enter" quanto clicando no botão correspondente, conforme imagens mais abaixo.
- 6. Montar dois painéis.
- 7. No painel da esquerda, listar os usuários filtrados.
- 8. No painel da direita, calcular e mostrar algumas estatísticas sobre esses usuários, conforme imagens abaixo.

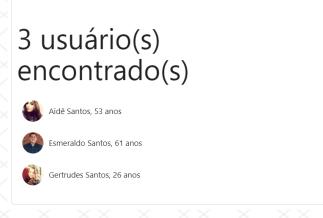
# Tela inicial da aplicação (sugestão):



### Tela após filtro "santos":

# Pesquisa de usuários

santos BUSCAR



# Estatísticas

Sexo mascuiino: 1 Sexo feminino: 2 Soma das idades: 140 Média das idades: 46,67

#### **Dicas**

- Após executar a requisição à API, obtenha somente os dados necessários ao app.
  Esses são: name (first + last), picture, dob.age e gender.
- 2. Monitore o input com o evento "keyup".
- 3. Filtrem os dados a partir de qualquer posição no nome, ou seja, o nome "Brenda" (caso exista na API) deve ser retornado se o filtro for "enda".
- 4. Para filtrar, considere todo o texto em minúsculas. Assim, o filtro "E" trará tanto "Elena" quanto "Helena", caso existam na API.
- 5. <u>Não faça a "limpeza" do texto, ou seja, devem ser considerados os acentos e caracteres especiais</u>. Portanto, o texto "Andre" <u>não pode filtrar</u> o nome "André".
- 6. Dê um console.log() nos dados do evento de digitação e você descobrirá como "cercar" a tecla "Enter".
- 7. Foque mais no código JavaScript e menos na interface. O mais importante é que o filtro e os cálculos estejam corretos.
- 8. Quebre o seu código em funções bem definidas.
- Será necessária uma boa dose de manipulação manual do DOM. Isso pode ser feito tanto com innerHTML + string (recomendo a utilização de template literals) ou com os comandos document.createElement, appendChild, etc.
- 10. Se quiser fazer uma interface semelhante à das imagens, utilize o Materialize (https://materializecss.com).
- 11. Não deixe de assistir o vídeo desse desafio, onde demonstro a aplicação em funcionamento e dou mais algumas dicas.

#### **Respostas Finais**

Os alunos deverão desenvolver a prática e, depois, responder às seguintes questões objetivas: