



escola
britânica de
artes criativas
& tecnologia

Profissão: Engenheiro Front-End



React Testing Library



GUIA DA AULA 3



Explore testes no React

● **Testing Library**

● **Função render**

● **Objeto screen**

● **FireEvent**



Acompanhe aqui
os temas que
serão tratados
na videoaula



Testing Library

O Jest é ótimo para escrevermos testes para aplicações que contenham apenas JavaScript, na sua forma mais pura, que não interage com o DOM.

Para os testes de componentes e aplicações React precisamos de mais uma ferramenta, a **React Testing Library** (RTL).



Função render

Com essa biblioteca podemos fazer com que o Jest saiba interpretar o DOM, renderize e interaja com os elementos React. Para renderizar um componente React com o React Testing Library precisamos utilizar a função **render**:

```

import {
  render,
  screen,
} from '@testing-library/react';
import Header from './componentes/Header';

test('Debe renderizar o header', () => {
  render(<Header />);
  expect(screen.getByText('EBAC'));
  .toBeInTheDocument(); // continuação da linha acima
});
  
```



Objeto screen

No exemplo, importamos a função `render` e o objeto `screen` da biblioteca `RTL`, com isso renderizamos o componente e depois utilizamos o objeto `screen` para localizar um elemento na tela, com o `.toBeInTheDocument()` criamos uma expectativa de que o texto “EBAC” no DOM.



FireEvent

Com o *screen*, podemos selecionar elementos tanto para criar expectativas nos testes ou selecioná-los para interagir com eles, por exemplo, digitar um valor. Para isso utilizamos o *screen* em conjunto com uma função chamada **fireEvent**:

```

import { render, screen, fireEvent } from '@testing-library/react';
import Contador from './componentes/Contador';

test('Deve contar até 1', () => {
  render(<Header />);
  fireEvent.click(screen.getByTestId('btn-contador'));
  expect(screen.getByText('1 clique(s))).toBeInTheDocument();
});
  
```



No exemplo, selecionamos o botão através de um atributo chamado **test-id**, com **getByTestId**, este é um atributo especial que o RTL utiliza e pode facilitar a recuperação de elementos.

```
<button type="button" data-testid="btn-contador">
```

```
+
```

```
</button>
```

Para conhecer a lista de eventos, acesse o link <https://testing-library.com/docs/dom-testing-library/api-events/>.

