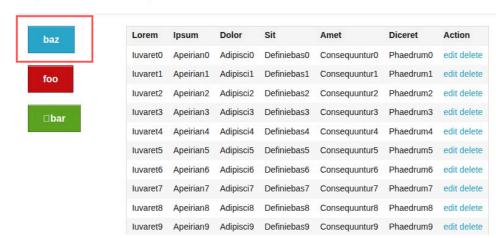
#### Melhores estratégias para mapeamento de botões

Link da página da tabela

#### 1. Primeiro Botão

## **Challenging DOM**

The hardest part in automated web testing is finding the best locators (e.g., ones that well named, unique, and unlikely to change). It's more often than not that the application you're testing was not built with this concept in mind. This example demonstrates that with unique IDs, a table with no helpful locators, and a canvas element.



O botão muda constantemente de id e de texto, por isso não é viável utilizá-los como identificadores. Para mapear o elemento de forma estável, foi utilizada a estratégia de XPath com base em um atributo fixo: a classe.

O botão é construído com a tag <a> (link) e possui a classe button, o que permite identificá-lo pelo seletor:

//a[@class='button']

# 2. Segundo Botão

# **Challenging DOM**

The hardest part in automated web testing is finding the best locators (e.g., ones that well named, unique, and unlikely to change). It's more often than not that the application you're testing was not built with this concept in mind. This example demonstrates that with unique IDs, a table with no helpful locators, and a canvas element.





Mesmas características do botão anterior, porém facilmente identificável pela classe 'button alert'.

O botão é construído com a tag <a> (link) e possui a classe button alert, o que permite identificá-lo pelo seletor:

//a[@class='button alert']

## 3. Terceiro Botão

# **Challenging DOM**

The hardest part in automated web testing is finding the best locators (e.g., ones that well named, unique, and unlikely to change). It's more often than not that the application you're testing was not built with this concept in mind. This example demonstrates that with unique IDs, a table with no helpful locators, and a canvas element.



Mesmas características do botão anterior, porém facilmente identificável pela classe 'button success'.

O botão é construído com a tag <a> (link) e possui a classe button success, o que permite identificá-lo pelo seletor:

//a[@class='button success']

4. Botão editar

## **Challenging DOM**

The hardest part in automated web testing is finding the best locators (e.g., ones that well named, unique, and unlikely to change). It's more often than not that the application you're testing was not built with this concept in mind. This example demonstrates that with unique IDs, a table with no helpful locators, and a canvas element.



O botão Edit redireciona para a URL

https://the-internet.herokuapp.com/challenging\_dom#edit.

Embora seja possível mapear o elemento pelo texto visível (edit), optou-se por utilizar o link do atributo href, pois ele é mais estável. Essa escolha evita falhas em casos onde o texto sofra pequenas variações, como o uso de letras maiusculas (edit → Edit)

//a[contains(@href,'#edit')]

#### 5. Botão deletar

## **Challenging DOM**

The hardest part in automated web testing is finding the best locators (e.g., ones that well named, unique, and unlikely to change). It's more often than not that the application you're testing was not built with this concept in mind. This example demonstrates that with unique IDs, a table with no helpful locators, and a canvas element.



O botão 'delete' possui as mesmas características do botão anterior, então foi usada a mesma estratégia.

//a[contains(@href,'#delete')]