

Mapeamento de Elementos HTML para Automação de Testes Web – Exercício Individual

Introdução

Este documento detalha o mapeamento de elementos HTML do site <https://alistamento.eb.mil.br/alistamento> com o objetivo de automatizar testes web. O mapeamento foi realizado utilizando diferentes estratégias de seletores e cada escolha é justificada.

Passo a Passo: Como encontrar elementos usando o DevTools

1. Acessar o site: <https://alistamento.eb.mil.br/alistamento>
2. Abrir as Ferramentas do Desenvolvedor: F12 ou clique com o botão direito e selecione Inspecionar.
3. Usar a ferramenta de seleção (ícone de seta no canto superior esquerdo da aba Elements) e clicar no elemento desejado.
4. Buscar seletor pelo Ctrl + F dentro da aba Elements, digitar o seletor CSS ou XPath e verificar se o elemento é encontrado corretamente.

Tabela de Mapeamento

A seguir, uma tabela com os elementos mapeados, seletores utilizados e justificativas.

Elemento	Seletor utilizado	Estratégia utilizada	Por que escolhemos essa estratégia	Como buscar com Ctrl + F
Menu de Navegação	nav#fixedTopNav	ID único	Navegação é essencial para testes de fluxo. Como o elemento possui ID fixo → usamos o ID direto.	nav#fixedTopNav
Seção de Conteúdo	section.section	Classe fixa	Representa o conteúdo principal da página → Usamos a classe fixa para garantir clareza e estabilidade.	section.section
Links dentro da Seção	section.section.container a	CSS combinado (hierarquia)	Seleciona todos os links dentro da seção principal, útil para garantir que os links estão corretos.	section.section.container a
Rodapé	footer#footer	ID único	Rodapé é	footer#footer

			importante para verificar informações estáticas. Como tem ID → usamos ID.	
Botão "Voltar ao Topo"	a#totop	ID único	Importante para testar rolagem e interação. ID fixo facilita o mapeamento.	a#totop
Links clicáveis dentro da seção	document.querySelector('section.section').querySelectorAll('a, button, input')	Seletores múltiplos (querySelectorAll)	Utilizamos múltiplos seletores CSS para capturar todos os elementos interativos de forma abrangente.	section.section a, button, input

Por que escolhi essas estratégias

- ID único: Sempre que o elemento possui um ID fixo e único, por ser mais simples e robusto.
- Classe fixa: Quando não há ID, mas o elemento é identificado por uma classe clara e semântica.
- CSS combinado (hierarquia): Para buscar elementos filhos dentro de seções específicas, ideal para capturar arrays de links ou botões
- querySelectorAll: Quando precisamos capturar **múltiplos tipos de elementos interativos de uma vez**
- XPath com texto: Não foi necessário nesse site, pois conseguimos mapear tudo com CSS e ID.