

# Lista de Checagem para Acessibilidade na Web

## Marcação

- Separe estrutura de apresentação e use marcação adequada para a estrutura. Exemplo: Use `<ul>` e `<ol>` para listas ao invés de um `<br>` após cada item.
- Cabeçalhos do HTML como `<h1>`, `<h2>`, `<h3>` e assim por diante são muito úteis para usuários sem visão. Marque adequadamente as seções da página e o corpo do documento com cabeçalhos semânticos ao invés de fazer outro elemento parecer um título por CSS.
- Dê títulos descritivos e com significado preciso para as páginas utilizando a tag `<title>`.
- Indique o idioma principal do documento usando o atributo `lang` na tag `<html>`, também o utilize em outras tags do documento caso apresentem conteúdo em idioma diferente do principal. Exemplo: `<span lang="es">Hola</span>` significa Olá.
- Forneça um link "Pular para o conteúdo" no topo do documento para que seja possível pular toda a navegação indo direto para o conteúdo.
- Sempre indique cabeçalhos em tabelas usando a tag `<th>`, e associe todas as células a seus respectivos cabeçalhos.
- Assegure-se de oferecer uma ordem lógica para a navegação com tecla tab utilizando o atributo `tabindex`. (Mas se seu código HTML já estiver na ordem adequada não é preciso utilizar este atributo.)

## Aparência Visual & Conteúdo

- Certifique-se que a página continua utilizável com as imagens desabilitadas. (Isso inclui verificar se o contraste continua suficiente para leitura sem as imagens de fundo.)
- Certifique-se que a página se mantém usável quando o usuário aumenta o texto até duas vezes o tamanho original.
- Certifique-se que cada elemento na página possa ser alcançado e controlado pelo teclado.
- Sempre que possível escreva cabeçalhos descritivos e textos de links que podem ser compreendidos fora do contexto (nada de links "clique aqui").
- Assegure-se que seu conteúdo tem bom contraste com o fundo, até para usuários daltônicos ou com pouca visão.
- Não use elementos que pisquem mais de três vezes por segundo.
- Não esconda o indicador de foco. Quando um usuário usar a tecla tab para navegar deve ficar aparente qual elemento está em foco.
- Não exija que os usuários percebam fontes, cores ou outros estilos visuais para entender o significado. Nada de "A palavra destacada no parágrafo anterior é a mais importante", ou "Itens marcados em vermelho são erros e precisam ser corrigidos", a menos que a palavra ou itens sejam indicados de algum outro modo.

## Conteúdo Dinâmico

- Não use eventos em JavaScript que alterem radicalmente a página ou carreguem uma página inteiramente nova quando acionados.

## Imagens e Multimídia

- Assegure-se que **todas** as imagens possuam o atributo `alt`, deixando o texto em branco caso a imagem seja apenas decorativa (ex: `alt=""`).
- Adicione o atributo `alt` mesmo que as imagens sejam também links.
- Seja breve com o conteúdo do atributo `alt` (ex: "Catedral de Notre Dame"), mas forneça detalhes quando eles derem significado (ex: "Filho em pé com sua mãe nos braços").
- Forneça transcrições, legendas e/ou tradução em linguagem de sinais para todo conteúdo de áudio e vídeo com falas.
- Forneça uma versão descrita de um vídeo quando a descrição é necessária para usuários sem visão entenderem o conteúdo. (O áudio descrito pode ser distribuído com o conteúdo do vídeo ou como um arquivo de áudio apenas.)
- Certifique-se que todos os vídeos, caso não iniciem automaticamente, tenham um controle de início acessível.
- Quando o texto pode ser renderizado pelo navegador tão bem como apresentado em uma imagem, evite usar imagens para o texto. (Técnicas de *Image Replacement* costumam ser uma alternativa aceitável, mas considere também as traduções quando utilizar texto em ou como imagens.)
- Evite *Captchas* a não ser que você não tenha alternativa, e mesmo assim ainda tente evitá-los. No entanto se você realmente não tem saída, forneça um *Captcha* alternativo em áudio.

## Formulários

- Sempre rotule **todos** os campos de formulários com a tag `<label>`. Se um campo do formulário não tiver texto específico de rotulo na página, adicione um, e esconda por CSS ou use o atributo `title`.
- Use áreas (`<fieldset>`) com legendas (`<legend>`) para associar com botões de seleção e caixas de checagem. Exemplo: Um formulário pergunta "Sexo:" e oferece os botões de seleção "Masculino" e "Feminino", "Sexo:" ficaria em uma tag `<legend>` e todos os três elementos (a tag `<legend>` e os dois botões de seleção) ficariam dentro da tag `<fieldset>`.
- Identificar todos os erros de valores que foram enviados (indo além de apenas imagens ou ícones), e colocar a notificação de erro próxima ao campo correspondente, ou em destaque no topo da página e com um link para o campo em questão.
- Forneça links de ajuda ou instruções sobre como preencher os campos quando necessário.
- Não permita que os usuários completem ações importantes sem confirmação ou possibilidade de desfazerem.
- Evite utilizar elementos do HTML de formas não semânticas, como elementos de formulários para navegação ou links para envio de formulários.

## Testando

- Valide sua marcação com o [validador do W3C](#). Se sua página não passar no teste pode existir uma boa razão para isso.
- Teste todas as páginas com simuladores de daltonismo. (Recomendados: [Color Oracle](#) e [Vischeck](#).)
- Teste todas as páginas com um [avaliador de acessibilidade](#), mas apenas após fazer tudo o que pode para assegurar a acessibilidade da página (como seguir estas recomendações).
- Tenha suas páginas analisadas por um perito em acessibilidade.

[Documento criado](#) por Aaron Cannon.

Tradução livre por [Emanuel Felipe](#).