

Enap Fundação Escola Nacional de Administração Pública

Enap Presidente

Enap Francisco Gaetani

Enap Diretor de Desenvolvimento Gerencial

Enap Paulo Marques

Enap Coordenadora-Geral de Educação a Distância

Enap Natália Teles da Mota Teixeira

Enap

Enap

Enap

Enap

Diagramação realizada no âmbito do acordo de Cooperação Técnica FUB/CDT/Laboratório Latitude e Enap.

Enap

© Enap, 2015

Enap

Enap Escola Nacional de Administração Pública

Enap

Diretoria de Comunicação e Pesquisa

Enap

SAIS - Área 2-A - 70610-900 — Brasília, DF

Enap

Telefone: (61) 2020 3096 - Fax: (61) 2020 3178

Enap

Enap

SUMÁRIO

1	Recomendações de acessibilidade	5
1.1	WCAG 2.0	5
1.2	eMAG.....	8
2	Práticas Web acessíveis.....	9
2.1	Marcação	10
2.2	Comportamento	17
2.3	Conteúdo/ Informação	20
2.4	Apresentação/Design	25
2.5	Multimídia	32
2.6	Links.....	35
2.7	Imagens.....	38
2.8	Tabelas.....	46
2.9	Formulários.....	50
2.10	Padrão eGov	58
3	Avaliação de Acessibilidade	62
4	Recursos e ferramentas	63
5	Glossário	64
6	Conclusão	65

Enap
Enap

Enap

Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap

1 Recomendações de acessibilidade

Definição

As recomendações ou diretrizes de acessibilidade são documentos que visam a tornar o conteúdo Web acessível a todas as pessoas, inclusive às pessoas com deficiência, destinando-se aos autores de páginas, projetistas de sítios e aos desenvolvedores de ferramentas para criação de conteúdo.

A observação dessas recomendações não beneficia somente pessoas com deficiências, mas, também, facilita o acesso ao conteúdo da Web, independentemente da ferramenta utilizada (navegadores Web para computadores de mesa, *laptops*, telefones celulares, ou navegador por voz) e de certas limitações técnicas, como, por exemplo, uma conexão lenta, a falta de recursos de mídia, etc.

O principal documento internacional de recomendações de acessibilidade é o WCAG 2.0 - Web Content Accessibility Guidelines, ou Diretrizes de Acessibilidade para Conteúdo Web. No Brasil, existe o eMAG - Modelo de Acessibilidade em Governo Eletrônico, que contém as recomendações de acessibilidade a serem seguidas nos sítios e portais do governo brasileiro.

1.1 WCAG 2.0

As WCAG (Web Content Accessibility Guidelines) são as recomendações de acessibilidade para conteúdo da Web, ou seja, são diretrizes que explicam como tornar o conteúdo Web acessível a todas as pessoas.



Web Content Accessibility Guidelines 2.0

Elas foram desenvolvidas pelo consórcio W3C - World Wide Web, por meio da WAI (Iniciativa de Acessibilidade na Web), em colaboração com pessoas e organizações em todo o mundo.

A versão 1.0 foi lançada em 5 de maio de 1999 e a versão 2.0 em 11 de dezembro de 2008.

Enap

Enap
Enap

Enap

Enap
Enap

Documentos da WAI - Web Accessibility Initiative:

- WCAG: para conteúdo Web.
- ATAG - Authoring Tool Accessibility Guidelines: para ferramentas de autoria, editores HTML, Content Management Systems (CMS), *blogs*, *wikis*, etc.
- UAAG - User Agent Accessibility Guidelines: para navegadores Web, *media players* e outros agentes de usuário.
- WAI-ARIA - Accessible Rich Internet Applications Suite: aplicações Web ricas e acessíveis (desenvolvidas com Ajax, por exemplo).

Estrutura

O documento WCAG 2.0 está estruturado em quatro princípios, cada qual contendo recomendações. As recomendações possuem critérios de sucesso que devem ser seguidos. Para seguir os critérios de sucesso, são disponibilizadas técnicas específicas.



4 princípios

- 1° Princípio: Perceptível - a informação e os componentes da interface do usuário têm de ser apresentados aos usuários em formas que eles possam perceber.
- 2° Princípio: Operável - Os componentes de interface com o usuário e a navegação têm de ser operáveis.
- 3° Princípio: Compreensível - A informação e a operação da interface com o usuário têm de ser compreensíveis.
- 4° Princípio: Robusto - O conteúdo tem de ser robusto o suficiente para poder ser interpretado de forma concisa por diversos agentes do usuário, incluindo recursos de tecnologia assistiva.

Recomendações pertencentes a cada princípio

1° Princípio: Perceptível

- 1.1 Fornecer alternativas textuais para qualquer conteúdo não textual;
- 1.2 Fornecer alternativas para multimídia;

- 1.3 Criar conteúdo que possa ser apresentado de modos diferentes sem perder informação ou estrutura;
- 1.4 Tornar mais fácil aos usuários a visualização e a audição de conteúdos, incluindo as separações das camadas da frente e de fundo.

2º Princípio: Operável

- 2.1 Fazer com que todas as funcionalidades estejam disponíveis no teclado;
- 2.2 Prover tempo suficiente para os usuários lerem e usarem o conteúdo;
- 2.3 Não projetar conteúdo de forma conhecida por causar ataques epiléticos;
- 2.4 Prover formas de ajudar os usuários a navegarem, e localizarem conteúdos e determinar onde estes últimos se encontram.

3º Princípio: Compreensível

- 3.1 Tornar o conteúdo de texto legível e compreensível.
- 3.2 Fazer com que as páginas da Web apareçam e funcionem de modo previsível.
- 3.3 Ajudar os usuários a evitarem e corrigirem erros.

4º Princípio: Robusto

- 4.1 Maximizar a compatibilidade entre os atuais e futuros agentes do usuário, incluindo os recursos de tecnologia assistiva.

Critérios de sucesso

Para cada recomendação, existem critérios de sucesso, que são pontos específicos que devem ser atingidos.

Exemplo:

- **Recomendação 3.3 - Assistência de Entrada: Ajudar os usuários a evitarem e corrigirem erros**
 - ✓ **Critério de sucesso 3.3.1 - Identificação do Erro:** Se um erro de entrada for automaticamente detectado, o item que apresenta erro é identificado e o erro é descrito ao usuário por texto (Nível A).

E para cada critério de sucesso estão disponíveis **técnicas específicas** com exemplos de como o objetivo do critério pode ser atingido e testado.

- **Cada critério de sucesso é indicado por um nível de conformidade, que pode ser A, AA ou AAA:**
 - ✓ Nível A: barreiras mais significativas de acessibilidade. Estar em conformidade apenas com os critérios de nível A não garante um sítio altamente acessível;
 - ✓ Nível AA: estar em conformidade com todos os critérios de sucesso de nível AA garante um sítio bastante acessível, ou seja, o sítio será acessível para a maioria dos usuários, sob a maior parte das circunstâncias e utilizando-se a maioria das tecnologias.
 - ✓ Nível AAA: o nível de conformidade triplo A é bastante meticuloso, ou seja, visa a garantir um nível otimizado de acessibilidade. A maioria dos critérios de sucesso

- ✓ Terceiro passo: Avaliação de Acessibilidade.
- ✓ Manutenção da acessibilidade.
- ✓ Recursos, ferramentas e leitura complementar.

- **Parte 3 - Recomendações de acessibilidade (divisão por Seções)**

- ✓ Marcação (9 recomendações)
- ✓ Comportamento (7 recomendações)
- ✓ Conteúdo / Informação (12 recomendações)
- ✓ Apresentação / Design (4 recomendações)
- ✓ Multimídia (5 recomendações)
- ✓ Formulários (8 recomendações)

Total: 45 recomendações

- **Parte 4 - Elementos padronizados de acessibilidade digital no Governo Federal**

- ✓ Padroniza elementos de acessibilidade que devem ter características em comum em todas as páginas do governo federal, como:
 - atalhos;
 - barra de acessibilidade;
 - mapa do sítio;
 - página de acessibilidade, entre outros.

- **Parte 5 - Práticas desaconselhadas**

- **Parte 6 - Glossário**

- **Saiba Mais sobre o eMAG**

- ✓ [Conheça o eMAG \(link para um novo sítio\)](#)

2 Práticas Web acessíveis

Tim Berners-Lee, diretor do W3C afirma que o poder da Web está na sua universalidade. Para ele:



Ser acessada por todos, independente de deficiência, é um aspecto essencial.



Nesta seção, vamos aprender como desenvolver um conteúdo Web mais acessível.

Seu código deve ser acessível ao maior número de usuários possível, incluindo:

- Usuários de leitor de tela;
- Quem navega somente pelo teclado;
- Quem utiliza navegador com JavaScript desabilitado;
- Usuários de dispositivos móveis;
- Quem utiliza outros recursos de Tecnologia Assistiva;
- Usuários inexperientes;
- Entre outros.

Enap
 Enap

Vamos começar?

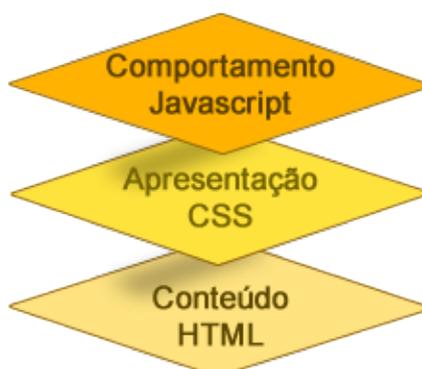
2.1 Marcação

Padrões Web e uso semântico do HTML

Quando falamos em Web acessível, é preciso ter em mente que o mais importante para a acessibilidade é o código HTML - HyperText Markup Language. O leitor de tela e outros recursos de Tecnologia Assistiva interpretam o código HTML, seus elementos e semântica. Por isso, o primeiro passo para garantir acessibilidade ao conteúdo Web é utilizar código semanticamente correto, ou seja, cada elemento para o seu propósito, seguindo-se os **Padrões Web ou Web Standards** do W3C - World Wide Web.

Os Padrões de Desenvolvimento Web, ou Web Standards, são um conjunto de normas, diretrizes, recomendações, notas, artigos, tutoriais e afins de caráter técnico, produzido pelo consórcio W3C, com a finalidade de atingir uma padronização e a criação de uma "Web universal". No [sítio do W3C \(link para um novo sítio\)](#), é disponibilizada uma divisão por assunto dos Padrões Web. Aqui, vamos focar os Padrões Web para desenvolvimento de sítios e, nesse contexto, o desenvolvedor precisa ficar atento aos seguintes conceitos:

- **Conceito de Camadas:** Representa o uso adequado das tecnologias, ou seja, cada tecnologia deve ser utilizada com o propósito único ao qual foi criada. Por exemplo, o HTML é utilizado apenas para a criação de conteúdo, enquanto a apresentação visual é criada por meio do CSS. Já o comportamento é definido pelas linguagens de *script*.



Enap

Enap
 Enap

- **Conceito de *Tableless*:** As tabelas devem ser usadas apenas para dados tabulares e não para fins de leiaute. O conceito de *tableless* surgiu devido ao uso excessivo de tabelas para apresentação visual de sítio. O uso de tabelas para a estruturação da página prejudica a apresentação da página em determinadas resoluções de tela, deixa a página mais pesada, tem como consequência um código de difícil manutenção, além de causar problemas de acessibilidade.

Leitura e tabulação

Deve-se criar o código HTML com uma sequência lógica de leitura para percorrer *links*, controles de formulários e objetos. Essa sequência é determinada pela ordem que se encontra no código HTML.

É recomendável que seja disponibilizado o bloco do conteúdo principal da página antes do bloco de menu. Isso facilita o acesso por quem navega via teclado, permitindo um acesso mais ágil, sem a necessidade de navegar por todos os itens de menu antes de chegar ao conteúdo. Apesar de os atalhos (como *links* e teclas) auxiliarem nesse sentido, alguns usuários não sabem como utilizá-los e eles podem ser de difícil utilização por pessoas com deficiência motora. Algumas das formas não funcionam em interfaces mais simples, como o WebVox do DOSVOX. Lembrando que, ao se utilizar CSS, visualmente, os blocos de menu e conteúdo podem ser dispostos em qualquer local da página.

Obs.: Em alguns casos, pode-se forçar a navegação via atributo *tabindex*. No entanto, se houver a necessidade de utilizar o *tabindex*, o mesmo deverá ser usado com a semântica correta e deverá ser verificado manualmente se o fluxo fornecido é realmente o desejado. O uso do *tabindex* pode resultar em uma ordem e tabulação inconsistente.

Quando pensamos em facilitar a leitura, especialmente para quem utiliza leitor de tela, é importante separarmos corretamente os *links* que estão em uma sequência (*links* adjacentes). Os mesmos devem ser separados por mais do que simples espaços, para que não fiquem confusos durante a leitura. Para isso, é recomendado o uso de **listas**, em que cada elemento dentro da lista é um *link*. As listas podem ser estilizadas visualmente com CSS para que os itens sejam mostrados da maneira desejada, como, por exemplo, um ao lado do outro.

Caso os *links* estejam no meio de um parágrafo de conteúdo de texto, podem-se utilizar vírgulas, parênteses, colchetes, *pipe*, entre outros, para fazer a separação.

Exemplo incorreto:

```
<p id="menu">
  <a href="#menu">Pular o menu</a><br />
  <a href="home.html">Home</a><br />
  <a href="pesquisa.html">Pesquisa</a><br />
  <a href="mapasítio.html">Mapa do Site</a>
</p>
<!-- Conteudo do Site -->
```

Exemplo correto:

```
<ul id="menu">
<li> <a href="home.html">Home</a></li>
<li> <a href="pesquisa.html">Pesquisa</a></li>
<li> <a href="mapasítio.html">Mapa do Site</a></li>
</ul>
<!-- Conteudo do Site -->
```

Níveis de título

Os cabeçalhos (títulos) em HTML são representados pelas *tags* <h1> até <h6> . Assim, as *tags* <h1>, <h2>, <h3>, <h4>, <h5> e <h6> informam ao navegador que aquele elemento trata-se de um cabeçalho (h vem do inglês "*heading*", que significa cabeçalho). O h1 deve ser o título principal da página, ou seja, deverá existir apenas um elemento h1 em cada página. A seguir, devem ser utilizados os níveis subsequentes de título, de maneira hierárquica, sem pular um ou mais níveis. A apresentação visual dos títulos poderá ser modificada por meio das folhas de estilo.

Para compreender melhor a utilização dos níveis de título, imagine o seu conteúdo como sendo um livro. O título do livro seria o h1, os capítulos seriam elementos h2, os subcapítulos seriam representados por elementos h3 e assim por diante. Dessa forma, o título principal de um sítio seria o h1, o que nos leva a ter apenas um h1 em cada página. O título do conteúdo principal passa a ser o h2, as seções internas do conteúdo seriam o h3 e assim por diante.

Mas por que é tão importante utilizar corretamente os títulos em um sítio? Para quem enxerga, basta diferenciar os títulos visualmente, pelo tamanho, tipo ou cor da fonte. No entanto, para quem não pode ver, essa diferenciação visual perde seu propósito. Pessoas que navegam pelos sítios utilizando um leitor de tela são informadas por esse *software* sobre o tipo de elemento pelo qual estão passando. Por exemplo, quando o usuário passar por um título dentro do conteúdo que foi inserido como um h2, o leitor de tela irá ler algo do tipo "cabeçalho de nível 2, texto do cabeçalho". Assim, o usuário, mesmo sem enxergar, saberá que se trata de um título e conhecerá a hierarquia do conteúdo da página. Além disso, os leitores de tela permitem que os usuários naveguem rapidamente pelos títulos, pressionando a tecla "H". Assim, esses usuários poderão pular de título em título, compreendendo a hierarquia da página e chegando rapidamente ao conteúdo desejado.

Enap
Enap

Enap

Enap
Enap

Exemplo:

Acessibilidade na Web

Introdução à Acessibilidade na Web

...texto...

Barreiras enfrentadas por pessoas com deficiência

...texto...

Recursos de Tecnologia Assistiva

...texto...

Garantindo a Acessibilidade na Web

...texto...

Recomendações de acessibilidade

...texto...

Avaliando a acessibilidade de um site

...texto...

Enap

```
<h1> Acessibilidade na Web </h1>
<h2> Introdução à Acessibilidade na Web </h2>
<p> ...texto...<p>
<h3>Barreiras enfrentadas por pessoas com deficiência</h3>
<p> ...texto...<p>
<h3>Recursos de Tecnologia Assistiva</h3>
<p> ...texto...<p>
<h2>Garantindo a Acessibilidade na Web</h2>
<p> ...texto...<p>
<h3>Recomendações de acessibilidade</h3>
<p> ...texto...<p>
<h3>Avaliando a acessibilidade de um sítio</h3>
<p> ...texto...<p>
```

Âncoras

As âncoras podem ser denominadas "*links* para uma mesma página", possibilitando aos usuários navegação por pontos específicos em uma página. As âncoras devem estar adequadamente descritas de acordo com sua função, e quando utilizadas devem ancorar o foco ao ponto indicado. Esse formato de *link* é geralmente empregado em casos em que o

Enap
Enap

Exemplo de âncora para a seção "Agradecimentos":

Repare que o destino da âncora é representado pelo sinal de # (cerquilha ou sustenido) seguido do id da seção de destino:

```
<li><a href="#agradecimentos" title="#">Agradecimentos</a></li>  
<h2 id="agradecimentos">Agradecimentos</h2>
```

Exemplo de âncora para voltar ao sumário:

```
<p class="voltar"><a href="#sumario">Voltar ao sumário</a></p>  
<div id="sumario">
```

Atalhos

Remetem a pontos estratégicos nas páginas, permitindo que o usuário navegue com maior agilidade, uma vez que ele poderá ir diretamente a pontos específicos das páginas. Os atalhos devem funcionar em todas as páginas do sítio, sendo indispensável disponibilizar dicas sobre eles no topo das páginas. Além disso, os atalhos devem estar adequadamente descritos, informando o local para o qual remetem. Devem existir atalhos para, no mínimo, ir ao início do conteúdo, ao início do menu e ao campo de busca. É importante que as dicas dos atalhos sejam apenas números, já que a tecla de ativação (Alt, Shift + Alt, etc.) varia conforme o navegador.

Enap

Exemplos de atalhos para conteúdo, menu e caixa de busca:



```
<ul id="atalhos">  
  <li><a href="#iniciodoconteudo">Conteúdo [1]</a></li>  
  <li><a href="#iniciodomenu">Menu [2]</a></li>  
  <li><a href="#palavra" id="teste">Pesquisar [3]</a></li>  
</ul>
```

Conteúdo da página:

```
<div id="conteudo">  
  <a href="#iniciodoconteudo" id="iniciodoconteudo" accesskey="1" class="oculto">Início  
  do conteúdo</a>  
  <!-- conteúdo -->  
</div>
```

Enap
Enap

Links Indicadores

São *links* associados aos atalhos de teclado, que agilizam a navegação, e auxiliam os usuários de leitores de tela a compreenderem e organizarem o conteúdo visual de forma lógica, pois definem blocos nas páginas. Os mais comuns e recomendados são "Início de Menu", "Fim de Menu", "Início de Conteúdo" e "Fim de Conteúdo", podendo também, se possível, ser inseridos *links* indicadores de início e fim de formulários e início e fim de tabelas. Os *links* indicadores poderão estar ocultos na página, ou seja, não aparecerão visualmente, mas serão lidos pelos leitores de tela, facilitando muito a navegação e a compreensão do sítio pelo usuário com deficiência visual.

Exemplo de *link* indicador de final de conteúdo:

```
<a href="#finaldoconteudo" class="oculto">Final do conteúdo</a>
```

2.2 Comportamento

Independência de dispositivo

Eventos que funcionam somente pelo *mouse* são um grande problema para a acessibilidade, pois nem todas as pessoas conseguem utilizar o *mouse*. Quem utiliza leitor de tela, por exemplo, navega exclusivamente pelo teclado. Pessoas com determinados tipos de deficiência física poderão não ter a habilidade de usar o *mouse* convencional e poderão utilizar teclados e *mouses* adaptados. Por isso, todas as funções de uma página devem ser dispositivo-independentes, ou seja, devem funcionar pelo *mouse*, teclado ou outro dispositivo. Além disso, o foco do teclado não deve ficar fixado ou "preso" em algum elemento, pois o usuário deve poder percorrer toda a página por meio do teclado facilmente.

Se forem utilizados eventos de JavaScript específicos para *mouse*, devem ser utilizados os correspondentes para teclado:

Eventos correspondentes para *mouse* e teclado

Evento do teclado	Evento correspondente do <i>mouse</i>
onkeydown	onmousedown
onkeyup	onmouseup
onkeypress	onclick
onfocus	onmouseover
onmouseover	onmouseout

Dê preferência a manipuladores de eventos que são dispositivo-independentes (funcionam pelo *mouse*, teclado ou outro), como: *onFocus*, *onBlur*, *onSelect*, *onChange*, e *onClick* (quando o *onClick* for utilizado em um link ou elemento de formulário).

Enap
Enap

Enap

Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap

Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap

Prefira utilizar o `onclick/onkeypress` em vez de `onmousedown/onkeydown` e `onmouseup/onkeyup`, pois estes últimos fazem com que o evento seja disparado automaticamente pelo teclado, o que é um grande problema para a acessibilidade.

Por fim, sempre teste o **sítio navegando por todo o conteúdo somente pelo teclado**, com setas e Tab, e utilizando em diferentes navegadores.

Objetos programáveis

Deve-se garantir que *scripts*, conteúdos dinâmicos e outros elementos programáveis sejam acessíveis e que seja possível a execução deles via teclado ou outro dispositivo. Um exemplo é a funcionalidade *drag and drop* (arrastar e soltar), que é utilizada por meio do *mouse*, mas pode ter como alternativa o uso de atalhos para recortar e colar, para quem utiliza exclusivamente o teclado.

Quando o *script* for utilizado em uma página da Web, uma forma de fornecer uma alternativa para ele é pelo elemento `NOSCRIPT`. Esse elemento pode ser utilizado para mostrar conteúdos em navegadores que não suportam *scripts* ou que tenham o *script* desabilitado. No entanto, se o navegador tiver suporte a *scripts* e estes estiverem habilitados, o elemento `NOSCRIPT` será ignorado. Dessa forma, a utilização do elemento `NOSCRIPT` para um *script* inacessível não garante que o objeto seja acessível. Assim, a recomendação é que o próprio *script* seja desenvolvido tomando-se o cuidado para que ele seja acessível, e o elemento `NOSCRIPT` deve ser utilizado para abranger os casos em que *scripts* não são suportados.

Em outras palavras, o elemento `NOSCRIPT` é uma alternativa para o *script*, mas não para a acessibilidade.

Enap

Exemplo:

Página HTML

```
<script src="js/exemplo.js" type="text/javascript"></script>
<noscript>Seu navegador não tem suporte a JavaScript ou está desativado!</noscript>
...
<a href="cadastro.html" id="cadastro">Cadastre-se agora!</a>
```

Arquivo JavaScript (exemplo.js)

```
function pop() {
  alert("Você vai fazer um novo cadastro!");
}
var element = document.getElementById("cadastro");
element.onclick = pop();
```

Quando não for possível que o elemento programável seja diretamente acessível, deve ser fornecida uma alternativa em HTML para o conteúdo.

Alterações temporais do conteúdo

É muito comum a utilização de conteúdos que se movem, como os *slideshows*. Esse tipo

Enap
Enap

seus dados. O sítio deve possuir um mecanismo que permita ao cidadão aumentar o limite de tempo, disponibilizando, por exemplo, um botão para aumentar o tempo em 5 minutos.

Intermitência de tela

Não devem ser utilizados efeitos visuais piscantes, intermitentes ou cintilantes. Em pessoas com epilepsia fotossensitiva, o cintilar ou o piscar podem desencadear um ataque epilético. Além disso, elementos piscantes podem causar desconforto, desatenção e confusão para os usuários.

2.3 Conteúdo/ Informação

Idioma

O idioma principal do sítio deve ser declarado em todas as páginas, já que auxilia na acessibilidade e melhora a indexação pelos motores de busca.

Exemplos:

Em HTML 4.01

```
<html lang="pt-BR">
<head>
<title>documento escrito em português do Brasil</title>
```

Em XHTML 1.1

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pt-BR" lang="pt-BR">
<head>
<title>documento escrito em português do Brasil</title>
```

Em HTML5

```
<html lang="pt-br">
<head>
<title>documento escrito em português do Brasil</title>
```

Quando houver mudança de idioma no conteúdo, o ideal é sempre identificar a mudança por meio do atributo "lang". Não há necessidade de informar a mudança de idioma para nomes próprios ou termos técnicos que sejam compreendidos no contexto.

Exemplos:

Em XHTML

```
<p xml:lang="de">
Da dachte der Herr daran, ihn aus dem Futter zu schaffen,
aber der Esel merkte, daß kein guter Wind wehte, lief fort
und machte sich auf den Weg nach Bremen: dort, meinte er,
könnte er ja Stadtmusikant werden.
</p>
```

Em HTML

```
<p lang="de">
Da dachte der Herr daran, ihn aus dem Futter zu schaffen,
aber der Esel merkte, daß kein guter Wind wehte, lief fort
und machte sich auf den Weg nach Bremen: dort, meinte er,
könnte er ja Stadtmusikant werden.
</p>

<p>
O leitor de tela <span lang="en">JAWS</span>
(acrônimo para <span lang="en">Job Access With Speech</span>)
é um software desenvolvido pela empresa americana
<span lang="en">FreedomScientific</span> e é, atualmente,
um dos mais populares no mundo.
Trata-se de um <span lang="en">software pago</span>,
desenvolvido para o sistema operacional
<span lang="en">Microsoft Windows</span>.
</p>
```

Título da página

O título da página é a primeira informação que será lida para quem utiliza leitor de tela, devendo ser descritivo e informativo, de modo que represente o conteúdo principal da página. O título é informado pelo elemento "title" e deve contemplar o assunto da página e o nome do sítio.

Exemplos:

Página inicial do Ministério da Educação:

```
<title>
Ministério da Educação
</title>
```

Enap

Enap

Enap

Se forem utilizados ícones para arquivos, é importante utilizar imagens e descrições de forma consistente dentro de um mesmo sítio, evitando confusão:

Utilização inconsistente

- Página 1



- Página 2



Utilização consistente

- Página 1



- Página 2



Texto

Acessibilidade também consiste em garantir que um texto possa ser compreendido por todos. Assim, o texto de um sítio deverá ser fácil de ler e compreender.

Enap
Enap

Enap

Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap

Algumas técnicas que podem auxiliar nesse sentido incluem:

- Desenvolver apenas um tópico por parágrafo;
- Utilizar sentenças organizadas de modo simplificado para o propósito do conteúdo (sujeito, verbo e objeto, preferencialmente);
- Dividir sentenças longas em sentenças mais curtas;
- Evitar o uso de jargão, expressões regionais ou termos especializados que possam não ser claros para todos;
- Utilizar palavras comuns no lugar de outras pouco familiares;
- Utilizar listas de itens em vez de uma longa série de palavras ou frases separadas por vírgulas;
- Fazer referências claras a pronomes e outras partes do documento;
- Utilizar, preferencialmente, a voz ativa.

Exemplo:

Repare no texto a seguir:

"Como exemplos de leitores de tela podemos citar o JAWS, que é um dos mais populares no mundo para Windows, mas é um *software* pago, o NVDA, o qual é gratuito e de código aberto para Windows, o ORCA, que também é gratuito e de código aberto, mas para Linux e o Virtual Vision, que é um leitor de tela pago para ambiente Windows, desenvolvido pela empresa brasileira MicroPower. Mais informações sobre esses leitores podem ser encontradas nos seus respectivos sítios: <http://www.freedomscientific.com/products/fs/jaws-product-page.asp> ([link para um novo sítio](#)), <http://www.nvda-project.org> ([link para um novo sítio](#)), <http://live.gnome.org/Orca> ([link para um novo sítio](#)) e <http://www.virtualvision.com.br> ([link para um novo sítio](#))."

O texto acima pode ser apresentado de maneira mais clara e fácil de compreender com base em pequenas modificações, como a utilização de listas de itens e a padronização da linguagem e da forma de apresentação:

"Como exemplos de leitores de tela, podemos citar:

- **JAWS:** *software* pago para Windows, sendo um dos mais populares pelo mundo – [sítio do Jaws \(link para um novo sítio\)](#);
- **NVDA:** *software* gratuito e de código aberto para Windows – [sítio do NVDA \(link para um novo sítio\)](#);
- **ORCA:** *software* gratuito e de código aberto para Linux – [sítio do Orca \(link para um novo sítio\)](#);
- **Virtual Vision:** *software* pago para Windows, desenvolvido pela empresa brasileira MicroPower – [sítio do Virtual Vision \(link para um novo sítio\)](#)."

Quando o texto incluir palavras incomuns, termos técnicos, palavras ou frases em outro idioma, etc., deverá ser fornecida uma explicação para tais palavras. Também deverão ser fornecidas as formas completas ou o significado de siglas e abreviaturas, ao menos na primeira vez em que aparecerem no texto.

As definições poderão ser fornecidas por meio de um texto adjacente, uma lista de definições, um glossário ou de qualquer outro modo.

2.4 Apresentação/Design

Contraste



Uma boa relação de contraste entre o texto e o plano de fundo é fundamental para que todos possam visualizar as informações de forma clara e sem grandes esforços. Além disso, um bom contraste é essencial para pessoas com baixa visão, pessoas com daltonismo e aquelas que utilizam monitores monocromáticos.

Além de escolher cores que tenham uma boa relação de contraste, é preciso tomar cuidado com imagens utilizadas de plano de fundo. Um fundo decorado ou em forma de figura, como uma paisagem, por exemplo, torna o conteúdo da página de difícil visualização, além de desviar a atenção do usuário. Se possível, evite utilizar imagens como plano de fundo ou planos de fundo decorados, enfeitados, com listras, círculos, bordas, marca d'água, etc.

A relação de contraste pode ser encontrada dividindo-se o valor da luminosidade relativa da cor mais clara de um dos planos pelo valor da luminosidade relativa da cor mais escura do outro plano. Levando-se em consideração a perda de percepção do contraste resultante da baixa acuidade visual, da cromodeficiência ou da perda de sensibilidade ao contraste devido ao envelhecimento, é recomendada, aqui, uma relação de contraste de, no mínimo, 4,5:1.

Enap

Contraste Baixo	Contraste Suficiente	Alto Contraste
Verde no cinza	Verde no cinza	Verde no cinza
Amarelo no azul	Amarelo no azul	Amarelo no azul

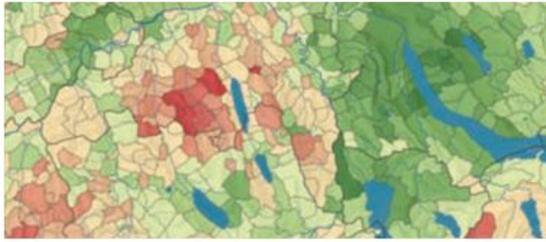
Apenas olhando para uma página, já podemos ter uma ideia se ela apresenta bom contraste ou não. No entanto, para garantirmos um contraste otimizado, podemos utilizar ferramentas gratuitas, disponíveis *online* para analisar a relação de contraste, como, por exemplo:

- [Luminosity Colour Contrast Ratio Analyser \(link para um novo sítio\)](#);
- [Color Contrast Analyser \(link para um novo sítio\)](#);
- [Check my colours \(link para um novo sítio\)](#).

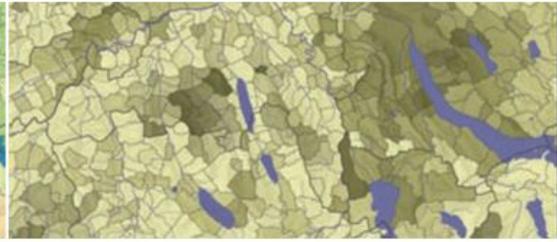
No "Check my colours" você fornece o endereço de uma página da Web, clica em "check" e ele mostra uma lista contendo o resultado das taxas de contraste entre o plano de fundo e o primeiro plano dos diversos elementos da página, conforme podemos ver nas figuras a seguir.

Enap

Visão Normal

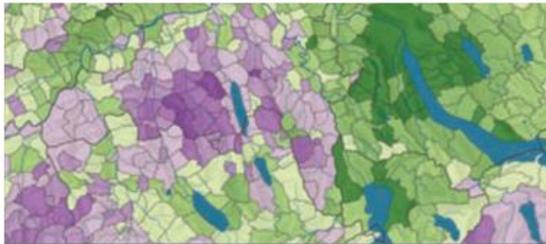


Deficiência de cores vermelha/verde

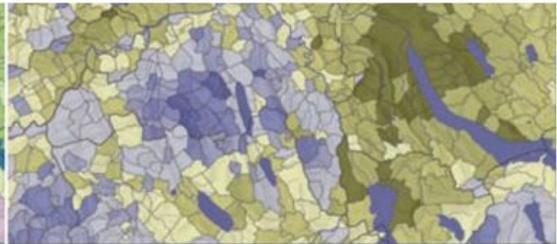


Utilizando combinação de cores verde/lilás: mesmo com a deficiência de cor, ainda é possível perceber o contraste.

Visão Normal



Deficiência de cores vermelha/verde



[Fonte: JENNY, B; KELSO, N.V. Color Design for the Color Vision Impaired](#)

Cor e outros elementos

As informações não podem ser transmitidas unicamente por meio de características sensoriais, tais como cor, forma, tamanho, localização visual, orientação ou som. Para compreendermos melhor esse ponto, vamos analisar alguns exemplos.

Exemplo 1:

Lista de produtos (os itens em vermelho não estão disponíveis):

- Produto 1.
- Produto 2.
- **Produto 3.**
- Produto 4.
- **Produto 5.**

Neste exemplo, é preciso que o usuário consiga enxergar e distinguir cores para saber quais produtos estão indisponíveis. Uma solução para esse caso seria, além de utilizar a cor como diferencial, informar ao lado do produto que ele encontra-se indisponível (**Produto 3 indisponível**).

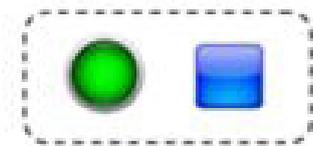
Exemplo 2:

O *link* para a página do concurso público está disponível no **canto superior direito da página**.

Repare que, no exemplo acima, é necessário que o usuário consiga visualizar uma posição específica da página. Nesse caso, o ideal seria fornecer o *link* no próprio contexto, facilitando o acesso por todos os usuários.

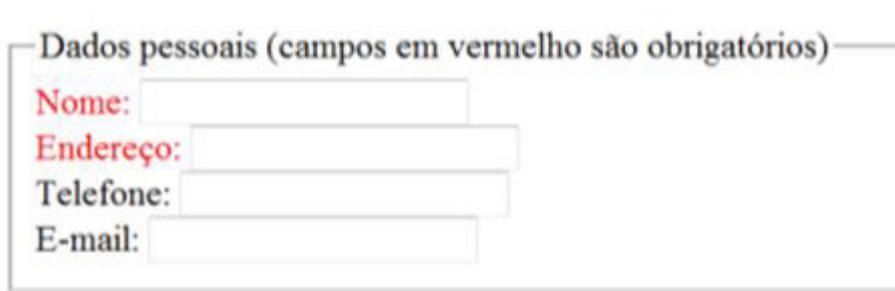
Exemplo 3:

Para salvar o questionário, clique no botão redondo. Para sair do questionário sem salvá-lo, clique no botão quadrado. Você poderá retornar mais tarde para preencher este questionário.



No exemplo acima, uma pessoa cega não saberia qual o formato dos botões e nem o leitor de tela teria como interpretar esse tipo de informação sensorial para transmitir ao usuário. Nesse caso, seria necessário fornecer informações adicionais aos botões, como uma descrição que identifique a sua função, ou então, o seu formato, de modo que o leitor de tela possa ler essa informação ao usuário.

Exemplo 4:



Enap

Repare que o formulário acima indica os campos obrigatórios apenas por meio da mudança de cor. Um usuário que não pode ver, não saberá quais campos são de preenchimento obrigatório.

Uma solução para o caso de campos obrigatórios em formulários seria inserir um asterisco em forma de imagem ao lado de cada campo obrigatório, fornecendo uma descrição no atributo "alt" para essa imagem, que poderia ser "campo obrigatório" ou, simplesmente, "obrigatório".

```
<label for="nome">Nome:
  
</label>
<input type="text" name="nome" id="nome" />
```

Para quem enxerga, basta visualizar o asterisco para distinguir os campos obrigatórios. Com o leitor de tela, a informação do "alt" seria lida para o usuário. No caso do exemplo de código acima, o leitor iria informar algo do tipo: "Campo de edição, nome, obrigatório".

Outra solução é utilizar uma *tag* "span" contendo a informação "obrigatório", que será ocultada por CSS e, em seu lugar, inserir a imagem de um asterisco.

```
<label for="nome">Nome:
  <span>Obrigatório</span>
</label>
<input type="text" name="nome" id="nome" />
```

Obs.: Mecanismos para ocultar elementos e seus efeitos na acessibilidade:

Formas de ocultar elementos e efeitos na acessibilidade

CSS	Efeito na tela	Efeito na acessibilidade
<code>visibility:hidden;</code>	O elemento fica oculto, mas continua a ocupar o espaço que normalmente ocuparia.	O conteúdo é ignorado pelos leitores de tela.
<code>display:none;</code>	O elemento fica oculto e não ocupa espaço.	O conteúdo é ignorado pelos leitores de tela.
<code>height: 0; width: 0; overflow: hidden;</code>	O elemento fica oculto e não ocupa espaço.	O conteúdo é ignorado pelos leitores de tela.
<code>text-indent: -999em;</code>	O conteúdo é movido para "fora da tela", não sendo mais visível, mas <i>links</i> podem ser focalizados de maneira imprevisível.	Os leitores de tela acessam o conteúdo, mas somente texto e elementos <i>inline</i> .
<code>position: absolute; left: -999em;</code>	O conteúdo é removido de sua posição, não ocupando espaço e é movido para "fora da tela", ficando oculto.	Os leitores de tela acessam o conteúdo.

Foco visível

Para pessoas com baixa visão, é muito importante que seja possível perceber facilmente onde está o foco do teclado, garantindo uma maior facilidade de navegação. Assim, a área que recebe o foco pelo teclado deve ser claramente marcada, e a área de seleção deve ser passível de ser clicada.

Por padrão, *links* e elementos de formulário já apresentam a borda destacada ao receberem o foco do teclado. Essa borda pode ser modificada via CSS para melhorar o destaque, mas não deverá ser removida. Recomenda-se que a espessura mínima da borda seja de 2px.

Exemplo:

CSS

```
a:focus, a:hover {
  border: 2px solid #F00;
}
```

A pseudo-classe `":focus"` é utilizada para definir o estilo de qualquer elemento HTML que receber o foco do teclado, como *links* e elementos de formulário. A pseudo-classe `":hover"` é utilizada para definir o estilo de um elemento quando se passa o *mouse* sobre ele.

Enap
Enap

Enap

Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap

HTML

```
<ul>
<li><a href="/acessibilidade/index.php">Página Inicial</a></li>
<li><a href="/acessibilidade/eventos.php">Eventos</a></li>
<li><a href="/acessibilidade/quemsomos.php">Quem Somos</a></li>
<li><a href="/acessibilidade/ead.php">Ensino a Distância (EaD)</a></li>
<li><a href="/acessibilidade/videoaulas.php">Vídeoaulas</a></li>
<li><a href="/acessibilidade/video.php">Vídeo em Libras</a></li>
<li><a href="/acessibilidade/oa.php">Objetos de Aprendizagem</a></li>
<li><a href="/acessibilidade/trabalhos.php">Trabalhos Realizados</a></li>
<li><a href="/acessibilidade/mapa.php">Mapa do Site</a></li>
</ul>
```



Redimensionamento

A página deve continuar legível e funcional mesmo quando redimensionada para até 200%. Assim, é preciso garantir que, quando a página for redimensionada, não ocorram sobreposições nem o aparecimento de uma barra horizontal.

Exemplo:

Sítio no qual, à medida que ocorre o redimensionamento, o leiaute se ajusta para que não ocorram "quebras".

Visualização padrão:



Visualização padrão da página web. O menu lateral contém itens como A INSTITUIÇÃO, ACESSO À INFORMAÇÃO, COLÉGIO DE DIRIGENTES, COMISSÕES, COMUNICAÇÃO, CONCURSOS, CONSELHO SUPERIOR, CONTATOS, CURSOS, ENSINO, EXTENSÃO, PESQUISA, PRONATEC e ADMINISTRAÇÃO. O conteúdo principal apresenta o artigo "Mestrado profissional em TIC prorroga inscrições", informando sobre a prorrogação do prazo de inscrições para o mestrado profissional do Programa de Pós-Graduação do Centro de Informática (Cin) da Universidade Federal de Pernambuco (UFPE) até 27 de fevereiro de 2014.

Visualização com redimensionamento de 200%:



Visualização da mesma página web com um zoom de 200%. Os elementos do menu lateral, o título do artigo e o texto do conteúdo são significativamente maiores em tamanho, mantendo a mesma disposição e estrutura visual.

Design Responsivo

Além de permitir o redimensionamento sem perda de funcionalidade, o leiaute do sítio deve adequar-se à resolução de tela do dispositivo pelo qual está sendo acessado, já que, atualmente, não existe mais um padrão de resolução de tela para os computadores e há uma crescente utilização de dispositivos móveis.

A melhor maneira de tratar as diferentes resoluções de tela é por meio do **design responsivo**, que trata especificamente da forma como o sítio será renderizado, dependendo do tamanho em *pixels* da tela que o dispositivo possui e seu objetivo. O *design* responsivo trata a resolução de tela considerando faixas de resolução, por diferentes folhas de estilo para tratamento das faixas de resolução de tela. Para tratar a folha de estilo, basta inserir no mesmo documento comandos de Media Queries contendo a resolução máxima de apresentação e as alterações desejadas logo depois.

Enap
Enap

Enap

Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap

Exemplo de leiaute responsivo, que se adapta conforme a resolução da tela:



Saiba Mais sobre design responsivo

- [O Cenário do Design Responsivo \(link para um novo sítio\).](#)



2.5 Multimídia

Ao disponibilizarmos um arquivo multimídia, é sempre importante garantir que todos possam ter acesso ao seu conteúdo, incluindo pessoas com deficiência visual e as com deficiência auditiva.

Para **vídeos que não possuem faixa de áudio**, é necessário fornecer alternativa em texto, ou seja, disponibilizar, junto ao vídeo, um arquivo para *download* ou um *link* para a transcrição textual.

A transcrição textual representa uma alternativa em texto que contém todo o conteúdo de um vídeo, incluindo tanto as informações contidas na faixa de áudio (se esta existir) quanto informações visuais transmitidas durante o vídeo. Assim, além das informações contidas nas falas, é preciso informar todo o conteúdo visual relevante para a compreensão do vídeo, como expressões corporais, risadas, informações em texto, mudança de ambiente, etc. A transcrição é uma alternativa para vídeos muito importante, pois possibilita o acesso a todo o conteúdo de um vídeo tanto para pessoas com deficiência visual quanto para pessoas com deficiência auditiva, além de pessoas com surdocegueira, que podem acessar o conteúdo da transcrição textual por um display Braille. Além de essencial para pessoas com deficiência visual e auditiva, a alternativa em texto também é importante para usuários que não possuem equipamento de som, que desejam apenas realizar a leitura do material ou não dispõem de tempo para assistir a um arquivo multimídia.

Para **arquivos apenas de áudio**, também é essencial fornecer um arquivo para *download* ou um *link* para a alternativa textual.



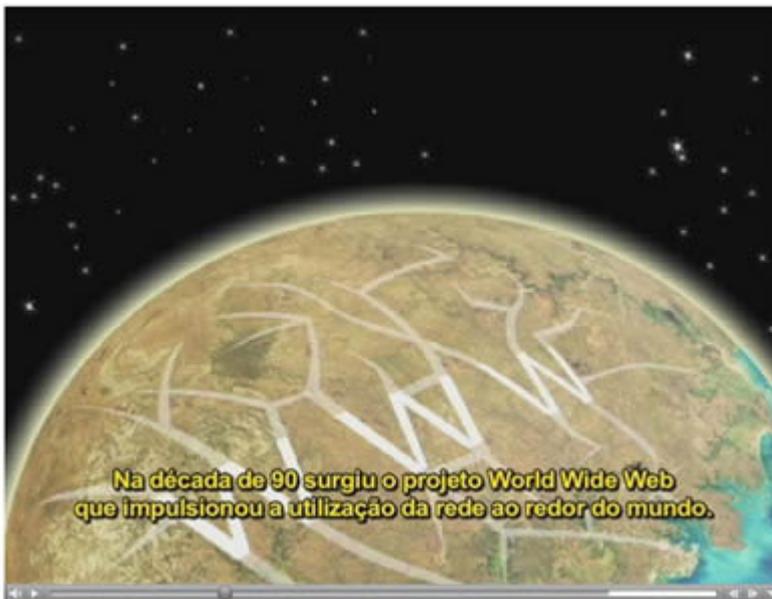
Transcrição textual do vídeo “Tableless - dicas de CSS”

[Expandir / Retrair](#)

Caso o **vídeo possua faixa de áudio**, além da alternativa em texto, devem ser fornecidas **legendas**. Se necessário, podemos fornecer, ainda, uma faixa de audiodescrição para o vídeo.

As legendas, textos sincronizados equivalentes ao conteúdo de áudio, são de fundamental importância para garantir a acessibilidade a pessoas com deficiência auditiva. Existem três tipos de legenda:

- Closed caption (CC): deverá ser ativada para começar a aparecer no vídeo;
- Open caption (OC): está sempre visível;
- Real time caption (legenda em tempo real): representam as legendas de vídeo ao vivo, ou seja, são criadas simultaneamente à apresentação do vídeo por pessoal especializado, utilizando *softwares* e equipamentos especiais.



Além das legendas, muitos vídeos irão exigir uma faixa de audiodescrição para que sejam totalmente acessíveis. A audiodescrição não é a transcrição textual do conteúdo de um vídeo, mas sim uma faixa de áudio que contempla informações que aparecem visualmente, mas não estão presentes nos diálogos ou no áudio do próprio vídeo. Assim, a audiodescrição permite que o usuário tenha acesso a informações visuais que transmitem conteúdo como expressões

Enap
Enap

Enap

Enap
Enap

Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap

faciais e corporais, informação sobre o ambiente, efeitos especiais, informações em texto que aparecem no vídeo, etc. A audiodescrição aparece no espaço entre as falas, sem sobrepor o conteúdo em áudio original do vídeo.

Acesse o [vídeo da Natura Mamãe e Bebê](#) para compreender melhor o que é a audiodescrição

Não podemos esquecer que é sempre desejável que os vídeos com áudio apresentem alternativa na Língua Brasileira de Sinais (Libras).



Biomias - Amazônia - Clube da Natureza (Laboratório de Tecnologia Educacional, Instituto de Biologia - UNICAMP)

Enap

Enap
Enap

É importante, também, que os arquivos multimídia não se iniciem automaticamente em uma página. Isto é, um vídeo, por exemplo, não deverá ser iniciado sem que o usuário tenha acionado um botão para esse fim. Dessa forma, para qualquer vídeo, áudio ou animação presentes na página, é necessário que sejam fornecidas ferramentas para que o usuário possa iniciar, pausar, reiniciar ou parar a apresentação desses elementos.

Para essa finalidade, o *player* de áudio ou vídeo deve ser acessível. Deve ser possível acessar os controles via teclado e os mesmos devem estar devidamente descritos, para serem interpretados pelos leitores de tela.



Resumo sobre elementos multimídia

- *Apenas áudio: Sempre fornecer transcrição textual;*
- *Áudio e vídeo: Sempre fornecer transcrição textual e legendas. Se necessário, fornecer faixa de audiodescrição. Se possível, fornecer alternativa em Libras;*
- *Todos os elementos (áudio, vídeo, animação): Não deverão ser iniciados automaticamente. O controle sobre os arquivos multimídia deve ser totalmente do usuário. Ele deverá poder iniciar, pausar, reiniciar ou parar a apresentação de determinado arquivo multimídia quando assim o desejar.*
- *O player deve ser acessível também por teclado e ter seus controles bem descritos para serem interpretados pelos leitores de tela.*



2.6 Links

Muitas pessoas utilizam apenas o *mouse* para navegar e nem imaginam que é possível navegar por meio da tecla "TAB" do teclado pelos elementos ativos de uma página. Isso inclui a navegação de *link* em *link*. Quando o usuário chega ao *link* desejado, basta ativá-lo com a ajuda da tecla "ENTER". Esse tipo de navegação é especialmente importante para pessoas que não conseguem utilizar o *mouse*, como é o caso de pessoas com problemas motores e de usuários com deficiência visual, que navegam utilizando um leitor de tela. É preciso considerar que, quando um usuário de leitor de tela navega pelos *links* de uma página, ele recebe apenas a informação do texto do *link* e não de todo o seu contexto. Por isso, o título do *link* deve descrever o seu destino. Veja o exemplo a seguir:

Notícias:

- Projeto arte na escola: coordenadora recebe prêmio. [Leia mais](#)
- Trabalho de academia de Engenharia Química é premiado em concurso. [Leia mais](#)
- Vestibular de verão está com inscrições abertas. [Leia mais](#)

No exemplo acima, o usuário, navegando de *link* em *link*, receberia a seguinte informação do leitor de tela: "Leia mais, *link*", "Leia mais, *link*", "Leia mais, *link*". O usuário provavelmente não acessaria esses *links*, pois não saberia qual o seu destino. Ele se perguntaria "Leia mais sobre o quê?". Uma solução nesse caso seria retirar o *link* "Leia mais" e transformar o próprio título da notícia em *link*, conforme o exemplo a seguir:

Notícias:

- [Projeto arte na escola: coordenadora recebe prêmio.](#)
- [Trabalho de academia de Engenharia Química é premiado em concurso.](#)
- [Vestibular de verão está com inscrições abertas.](#)

Desse modo, o texto de cada um dos *links* torna-se suficientemente claro e descritivo. Quando o usuário de leitor de tela passar pelo primeiro *link*, por exemplo, receberá a informação "Projeto Arte na Escola: coordenadora recebe prêmio, *link*". Assim, ele saberá que se trata de um *link* e saberá qual o seu destino, podendo decidir se deseja ou não acessá-lo.

Da mesma forma que o "Leia mais" é problemático para navegação com leitor de tela, *links* dos tipos "Clique aqui" e "Saiba mais" também são, pois não descrevem o destino do *link*. Assim, esse tipo de *link* devem ser evitados.

A maioria dos leitores de tela possui uma tecla de atalho que permite ao usuário acessar a lista de todos os *links* da página. É uma maneira fácil e rápida de ter acesso aos *links*. No entanto, se o texto dos *links* não for suficientemente descritivo, essa funcionalidade terá pouco valor. No exemplo a seguir, podemos ver uma parte da lista de *links* de uma página. Repare que existem vários *links* com o título "Leia Mais". O usuário irá navegar por esses *links* e irá se perguntar "leia mais sobre o quê?". O ideal para tornar esses *links* acessíveis é lhes fornecer um texto descritivo, como, por exemplo, o próprio título da notícia ou, até "leia mais sobre acessibilidade".

CSS:

```
a:active, a:focus, a:hover {
    border: 1px solid #F00;
}
```

HTML:

```
<ul>
  <li><a href="/acessibilidade/index.php">Página Inicial</a></li>
  <li><a href="/acessibilidade/eventos.php">Eventos</a></li>
  <li><a href="/acessibilidade/quemsomos.php">Quem Somos</a></li>
  <li><a href="/acessibilidade/ead.php">Ensino a Distância (EaD)</a></li>
  <li><a href="/acessibilidade/videoaulas.php">Vídeoaulas</a></li>
  <li><a href="/acessibilidade/video.php">VÍ-deo em Libras</a></li>
  <li><a href="/acessibilidade/oa.php">Objetos de Aprendizagem</a></li>
  <li><a href="/acessibilidade/trabalhos.php">Trabalhos Realizados</a></li>
  <li><a href="/acessibilidade/mapa.php">Mapa do Site</a></li>
</ul>
```



A pseudo-classe ":focus" é utilizada para definir o estilo de qualquer elemento HTML que receber o foco do teclado, como *links* e elementos de formulário. A pseudo-classe ":hover" é utilizada para definir o estilo de um elemento quando se passa o *mouse* sobre ele. Já a pseudo-classe:active define o estilo de *link* ativo. No caso do exemplo acima, a pseudo-classe ":active" foi utilizada para definir o estilo do *link* com foco para navegadores Internet Explorer 7 e anteriores, que ainda não tinham suporte à pseudo-classe ":focus".

Enap
Enap

Enap

Enap
Enap



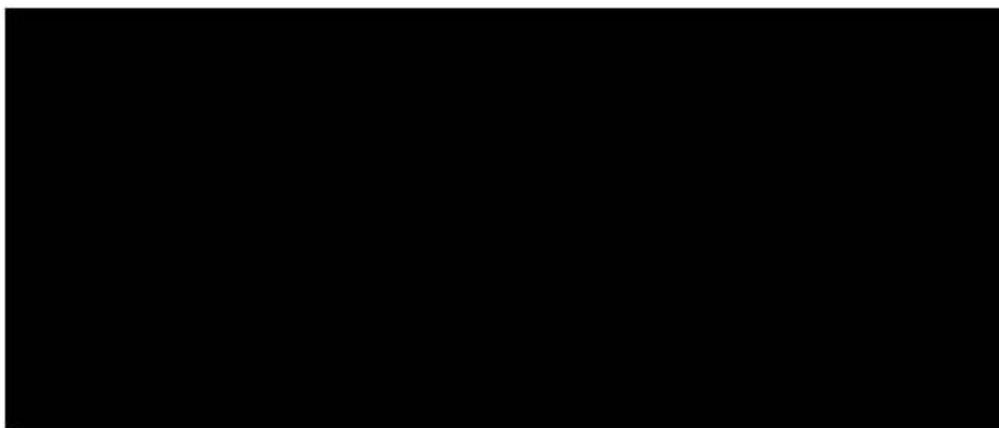
Resumo sobre links

- O texto do link deve ser claro e descritivo;
- O link deve ser compreensível mesmo fora de contexto;
- Evitar links do tipo "clique aqui", "leia mais", "saiba mais", etc.;
- Evitar abertura de nova aba ou janela;
- Informar a extensão e o tamanho do arquivo no texto do link para download;
- Permitir que o elemento com foco seja percebido visualmente.



2.7 Imagens

A imagem abaixo ilustra o organograma da instituição:



Você conseguiu compreender a imagem?

No contexto acima, se faltar a imagem, não conseguiremos compreender a informação. O mesmo ocorre no caso de pessoas cegas tentando acessar o conteúdo presente em imagens que não foram descritas.

Pessoas cegas não terão acesso ao conteúdo transmitido por uma imagem caso não tenha sido fornecida uma descrição para ela. Dessa forma, é preciso descrever de maneira apropriada todas as imagens que transmitem conteúdo.

Uma imagem pode ou não transmitir uma informação. Podemos tratar as imagens inseridas em uma página dividindo-as nos dois contextos abaixo:

- **Imagens com conteúdo:** fotos, gráficos, organogramas, ilustrações, imagens que substituem botões ou *links*, etc.

As imagens que apresentam conteúdo precisam de uma descrição. A descrição da imagem pode ser informada no próprio texto ou, então, por um atributo específico do elemento "img" no HTML, o atributo **alt**. Essa descrição fornecida não aparece visualmente, mas é lida pelo leitor de tela quando o usuário, navegando pelo teclado, passa pela imagem. Assim, mesmo que os usuários com deficiência visual não enxerguem a imagem, será informada a eles a sua descrição.

- **Imagens decorativas:** utilizadas para decorar a página ou partes dela. Como exemplos, podemos citar marcadores de lista estilizados, cantos arredondados, ícones utilizados para decorar ou enfatizar um *link*, etc.
- As imagens decorativas devem ser ignoradas pelos recursos de tecnologia assistiva, ou seja, o ideal é que os leitores de tela ignorem a presença dessas imagens, já que elas não transmitem conteúdo. Para que isso ocorra, a maneira mais indicada é **inserir todas as imagens decorativas por meio das folhas de estilo (CSS)**. Como alternativa, pode-se deixar o conteúdo do atributo "alt" vazio (alt=""), para que a imagem seja ignorada pelos leitores de tela.

Exemplos de descrição de imagens



```

```



```

```



```

```

Enap
Enap

Enap

Enap
Enap

Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap



``



Mas qual o limite para uma descrição ser considerada curta e poder ser disponibilizada por meio do atributo "alt"?

Não existe uma restrição oficial para o tamanho do texto alternativo fornecido no "alt", mas especialistas recomendam que ele seja inferior a 125 caracteres, aproximadamente.

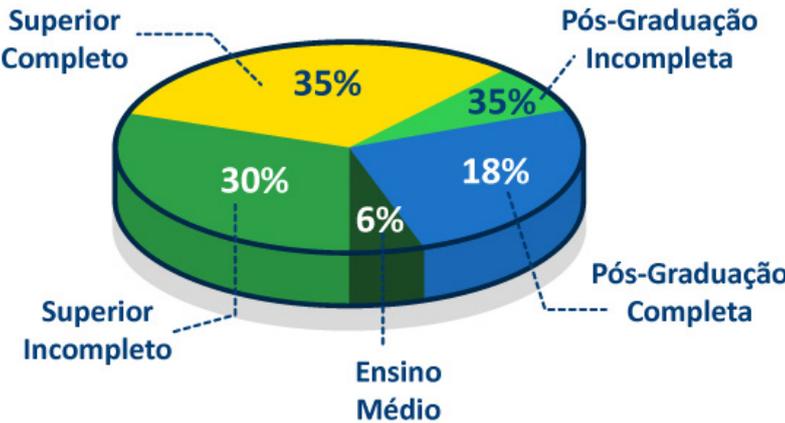
Enap



Enap
Enap

Imagens complexas

O exemplo a seguir apresenta um gráfico que foi inserido em forma de imagem. Quem não pode ver, não terá acesso às suas informações. Nesse caso, primeiramente, precisamos fornecer uma alternativa textual para a imagem no atributo "alt". No entanto, esse atributo é utilizado para descrições sucintas, em poucas palavras ou em uma frase curta. No caso de gráficos e imagens complexas, precisamos fornecer uma descrição mais detalhada. A solução, nesses casos, é fornecer as informações do gráfico no próprio contexto, antes ou depois da imagem ou então em um *link*, logo após a imagem. Conforme podemos ver no exemplo abaixo, foi fornecida uma breve descrição para a imagem no atributo "alt" e, além disso, foi fornecido o *link* para a descrição detalhada do gráfico. Nesse *link*, o usuário terá acesso a dados do gráfico em forma textual.



```

<p><a href="graficoPesquisa.html">A descrição textual do gráfico</a> está disponível
em outra página</p>
```

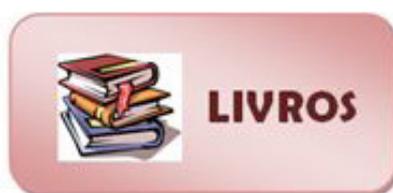
Ao desabilitarmos as imagens do navegador, podemos visualizar os textos alternativos:



Imagens que são *links*

Quando um usuário com leitor de tela passa por um *link*, ele será informado pelo leitor que aquele elemento é um *link* e será informado também o texto do mesmo. No entanto, se quisermos substituir um *link* por uma imagem, o ideal é que o façamos pelo CSS, de modo que o texto do *link* continue sendo lido normalmente pelo leitor de tela.

Caso não seja possível substituir o *link* pela imagem por meio do CSS, é necessário que a imagem possua um texto alternativo (alt). É preciso que o texto alternativo seja curto e objetivo, representando o seu destino. No exemplo abaixo, o texto no atributo "alt" deve ser simplesmente "livros", o mesmo texto que aparece visualmente na imagem.



```
<a href="livros.html">
  
</a>
```

Imagens que são botões

São elementos **input** do tipo **image** (`input type="image"`). Para os botões de imagem, que servem para o mesmo propósito do botão do tipo "submit", deve ser fornecida uma descrição textual por meio do atributo "alt", conforme o exemplo a seguir.



```
<input type="image" name="entrar" src="entrar.jpg" alt="entrar" />
```

Enap
Enap

Enap

Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap

Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap

Já para outros tipos de botões ("reset", "submit" e "button"), é preciso substituir o botão pela imagem que se deseja utilizar por meio do CSS, já que o comportamento dos leitores de tela quanto ao atributo "alt", nesses tipos de botões, é bastante variável. Nesse caso, para que o botão seja acessível, ele deve possuir um **value** descritivo, conforme o exemplo a seguir.



HTML:

```
<input type="reset" name="limpar" value="Limpar" class="btLimpar"/>
```

CSS:

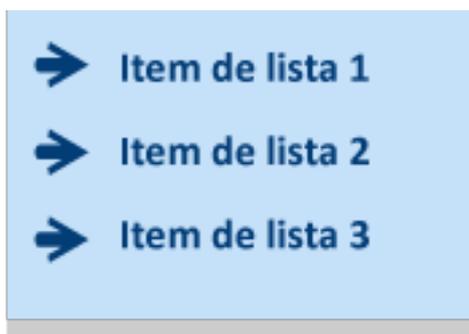
```
input.btLimpar{  
    background:transparenturl(btLimpar.jpg) no-repeat left top;  
    width:100px;  
    height:47px;  
    text-indent:-20000px;  
    border:0;  
}
```

Enap

Imagens decorativas

A ideia é que as imagens decorativas sejam ignoradas pelos leitores de tela. Para tal, essas imagens devem ser inseridas por **CSS**.

Um exemplo de imagem decorativa são os marcadores estilizados para listas (na figura a seguir, as setas). No exemplo abaixo, essas imagens foram inseridas por CSS.



HTML:

```
<ul class="bullet" >  
    <li>Item de lista 1</li>  
    <li>Item de lista 2</li>  
    <li>Item de lista 3</li>  
</ul>
```

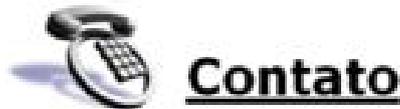
Enap
Enap

CSS:

```
#content ul.bullet li {
    list-style-type: none;
    text-align: center left;
    background-image: url(../images/ex-list-bullet.PNG);
    background-position: center left;
    background-repeat: no-repeat;
    padding: .3em 0 .2em 1.1em;
}

#content ul.bullet {
    margin: 0 0 .3em -.5em;
}
```

Como segunda opção, imagens sem conteúdo que não foram inseridas por CSS devem ter o atributo "alt" nulo (alt="") e não apresentar atributo **title**. O atributo "alt" com valor nulo indica ao leitor de tela que a imagem em questão pode ser ignorada. No exemplo a seguir, temos um *link* que é formado pelo texto do *link* ("Contato") e por um ícone (imagem de um telefone). Essa imagem está inserida no código dentro da *tag* <a>. O texto do *link* é "contato", informação essa que é suficientemente descritiva. Se fosse fornecida uma descrição para a imagem, o leitor de tela iria "ler" para o usuário informações desnecessárias. Assim, para que o leitor de tela ignore a imagem e leia simplesmente o texto do *link*, deixamos o "alt" da imagem nulo. Vale lembrar que o mais recomendado é inserir a imagem por CSS.



```
<a href="contact.html">
    
    Contato
</a>
```

Mapas de imagem

Um mapa de imagem é uma imagem dividida em áreas selecionáveis. Cada área é um *link* para outra página ou para outra parte da mesma página. Um exemplo de um mapa de imagem seria a imagem de uma estante de livros em que cada prateleira seria um *link* para um tipo específico de literatura.

Para que um mapa de imagem seja acessível, é preciso fornecer uma alternativa em texto para cada uma de suas áreas selecionáveis.

Enap
Enap

Enap

Enap
Enap

DICA

Quando for descrever uma imagem, pergunte-se: "se eu não pudesse utilizar esta imagem aqui, o que eu escreveria em seu lugar?"

- **Seja objetivo:** apesar de parecer uma boa ideia descrever uma imagem nos mínimos detalhes, na maioria das vezes, não existe essa necessidade. Busque informar simplesmente o que ela transmite, nem a menos, nem a mais. Deixe de lado detalhes pouco importantes e evite informar no texto alternativo mais do que a imagem está passando visualmente. Veja o exemplo a seguir:



Esta imagem tem a seguinte descrição:

```

```

Repare que, apenas olhando para a foto, a maioria das pessoas não saberia onde fica essa praia. Ou seja, o texto alternativo está informando mais do que a própria imagem passa visualmente. Um exemplo de texto alternativo apropriado para essa imagem seria:

```

```

- **Seja claro:** tome cuidado para não ser tão objetivo a ponto de seu texto não ficar claro o suficiente. Veja o exemplo a seguir:



Enap
Enap

Enap

Enap
Enap

Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap
Enap

- ✘ Descrição muito objetiva e pouco clara: alt="árvores" ou alt="floresta"
 - ✘ Descrição pouco objetiva: alt="eucaliptos com caules muito longos e folhas verdes plantados em fileira e fazendo sombra sobre a grama"
 - ✓ **Descrição objetiva e clara:** alt="plantação de eucaliptos"
- **Considere o contexto:** Procure não repetir na descrição da imagem informações que já estão no contexto, evitando que o leitor de tela leia duas vezes a mesma informação. No exemplo abaixo, não é necessário descrever as informações sobre inscrições e provas divulgadas na imagem, já que essa informação já está no contexto. Poderíamos inserir no "alt" apenas a informação "Banner do processo seletivo 2014/1". No entanto, se essa informação não estivesse no contexto, teríamos que descrever as informações de inscrições e provas.

Processo Seletivo Unificado 2014/1



As inscrições para o processo seletivo unificado 2014/1 vão de 7 de outubro a 11 de novembro. As provas serão realizadas no dia 1º de dezembro de 2013, nos municípios dos 12 câmpus da instituição.

2.8 Tabelas

Enap
Enap
Enap

As tabelas são utilizadas com a finalidade de disponibilizar dados tabulares, ou seja, para organizar informações em linhas e colunas, e não devem ser usadas para diagramação dos elementos visuais das páginas (*design*). O uso de tabelas com o propósito de diagramação não influencia apenas no nível de acessibilidade de um sítio, mas também na agilidade e no desempenho no carregamento de suas páginas. Logo, para estilizar e organizar os aspectos visuais das páginas, é recomendado se utilizarem as folhas de estilo (CSS).

Para pessoas que não possuem deficiência visual, a leitura e a compreensão das informações contidas nas tabelas ocorrem de forma simples, por meio do cruzamento visual de linhas e colunas, o que permite encontrar rapidamente os dados desejados. Já para usuários de leitores de tela, a obtenção e a compreensão dos dados contidos em tabelas não são uma tarefa tão fácil, especialmente se tiverem sido desenvolvidas de forma inacessível.

Sempre que possível, é recomendado que sejam utilizadas tabelas simples, evitando-se fazer uso de colunas e linhas mescladas ou de células em branco, pois isso dificulta a leitura da tabela por *softwares* leitores de tela, uma vez que os mesmos realizam a leitura de forma linear, ou seja, linha a linha. Além disso, alguns elementos e atributos são essenciais para a acessibilidade de tabelas, e devem ser empregados de forma lógica e semântica, de modo que venham a cumprir seu real propósito. A seguir, vamos conhecer como podemos implementar a acessibilidade em tabelas.

Exemplo:

```
<table summary="Material escolar - Levantamento comparativo de preços.">
  <caption>Material escolar</caption>
  <tr>
    <th>Estojo caneta hidrocor / loja</th>
    <th>6 unidades</th>
    <th>12 unidades</th>
    <th>24 unidades</th>
    <th>36 unidades</th>
  </tr>
  <tr>
    <th>Lisboa papelaria</th>
    <td>R$ 6,20</td>
    <td>R$ 9,99</td>
    <td>R$ 11,90</td>
    <td>R$ 16,90</td>
  </tr>
  <tr>...segue o resto da tabela...
```

O uso correto dessas marcações permite que os leitores de tela informem aos usuários sobre a função de cada elemento. Dessa forma, o usuário saberá o que é um cabeçalho, o que é uma linha e o que é uma célula de dados.

Elementos "thead", "tbody" e "tfoot"

Devem ser utilizados para tabelas extensas, possibilitando o agrupamento de linhas em uma tabela:

- "thead": aplicado para o agrupamento de linhas de cabeçalho;
- "tbody": usado para definir o corpo da tabela;
- "tfoot": utilizado para agrupar linhas de rodapé, demarcando o final de uma tabela.

Exemplo:

```
<table summary="....">
  <caption>....</caption>
  <thead>
    <tr> ...informação contida no cabeçalho... </tr>
  </thead>
  <tfoot>
    <tr>...informação contida no rodapé... </tr>
  </tfoot>
  <tbody>
    <tr> ...primeira linha de dados do bloco um... </tr>
    <tr> ...segunda linha de dados do bloco um...</tr>
  </tbody>
</table>
```

Segundo recomendações do W3C, é sugerido que o "tfoot" seja declarado no código antes do elemento "tbody", permitindo que os agentes de usuário renderizem os resultados da tabela antes de todas as possivelmente numerosas linhas do conteúdo. O emprego dessas marcações de agrupamento permite a fixação das linhas de cabeçalho e rodapé no navegador, possibilitando que o conteúdo ou corpo (tbody) permaneçam em movimento entre os limites demarcados. No entanto, esses elementos só irão funcionar caso possuam o mesmo número de colunas.

Associar células de dados a células de cabeçalho

Para tabelas mais complexas, é muito importante utilizar marcações para associar as células de dados com as células de cabeçalho. Essas associações podem ser realizadas por meio de:

- **Atributos "id" e "headers"**: o atributo "headers" é utilizado nas células de dados (<td></td>) em conjunto com o atributo "id" na célula de cabeçalho (<th></th>) para associar as células de dados aos seus respectivos cabeçalhos.

Exemplo:

```
<table summary="Oferta de cursos - tabela estruturada em: nome do curso, professor,
resumo da disciplina e carga horária">
  <caption>Oferta de cursos - Primeiro semestre</caption>
  <tr>
    <th id="t1" abbr="Nome">Nome do Curso</th>
    <th id="t2">Professor</th>
    <th id="t3">Resumo</th>
    <th id="t4" abbr="carga">Carga horária</th>
  </tr>
  <tr>
    <td header="t1">Gravura - conceitos básicos</td>
    <td header="t2">Eunice Lobato</td>
    <td header="t3">História da gravura, estilos, gráfica. Visita aos ateliês.</td>
    <td header="t4">40h</td>
  </tr>
  <tr>
    <td header="t1">Gravura em metal</td>
    <td header="t2">Ione Rocha</td>
    <td header="t3">Trabalhos práticos de gravura em metal. </td>
    <td header="t4">160h</td>
  </tr>
  <tr>
    <td header="t1">Xilogravura</td>
    <td header="t2">Heli Santos</td>
    <td header="t3">Trabalhos práticos em xilogravura.</td>
    <td header="t4">160h</td>
  </tr>
</table>
```

Enap
Enap

Enap

Enap
Enap

Utilização do label

Uma das maneiras mais comuns de navegação entre os campos de formulário é por meio da tecla "Tab". O usuário preenche um campo de formulário, pressiona a tecla "Tab", preenche o próximo campo, e assim por diante até chegar ao final do formulário. Para pessoas que enxergam, essa é uma tarefa bastante simples. No entanto, uma pessoa cega, que está navegando utilizando um leitor de tela, por exemplo, irá passar de campo em campo com a tecla "Tab" sem saber o que fazer nesses campos, a não ser que o formulário tenha sido desenvolvido de maneira que cada campo esteja relacionado a sua informação em texto correspondente. Para realizar tal associação, é utilizado o elemento "label".

As etiquetas de texto ou rótulos ("label") devem estar associadas aos seus devidos campos de formulário ("input") pelo uso dos atributos "for" contidos no elemento "label" e do atributo "id" do "input", sendo necessário atribuir o mesmo valor para ambos os atributos para que a associação seja válida.

A tag "label" pode ser usada para todos os elementos de formulário, exceto para os elementos "button" (seu funcionamento independe de *links* externos).

Exemplo:

```
<label for="nome">Nome: </label>
<input type="text" name="nome" id="nome" />
<label>Sexo:</label>
<input type="radio" id="fem" name="sexo" />
<label for="fem">Feminino</label>
<input type="radio" id="mas" name="sexo" />
<label for="mas">Masculino</label>
<label for="msg">Mensagem: </label>
<textarea name="msg" id="msg">Digite sua mensagem</textarea>
<input type="checkbox" id="receber" name="receber" />
<label for="receber">Deseja receber nossa newsletter?</label>
```

Para campos que exigem entrada de dados por parte do usuário, devem ser oferecidas, quando necessário, dicas de preenchimento junto aos rótulos dos campos ("label").

Agrupamento de campos

Visando à melhor organização do formulário, é recomendado o agrupamento de campos que possuem características comuns e que estão relacionados de maneira lógica, utilizando-se o elemento "fieldset", juntamente com o elemento "legend". Para cada um dos agrupamentos ("fieldset"), é possível fornecer uma legenda ("legend"), que deve explicar clara e objetivamente a finalidade ou a natureza dessas uniões.

Campos obrigatórios

Dados pessoais (campos em vermelho são de preenchimento obrigatório)

Nome:

Endereço:

Telefone:

E-mail:

Repare que o formulário acima indica os campos obrigatórios apenas por meio da mudança de cor. Um usuário que não pode ver não saberá quais campos são de preenchimento obrigatório.

Uma solução para o caso de campos obrigatórios em formulários seria inserir um asterisco em forma de imagem ao lado de cada campo obrigatório, fornecendo uma descrição no atributo "alt" para essa imagem, que poderia ser "campo obrigatório" ou, simplesmente, "obrigatório".

```
<label for="nome">Nome:  
    
</label>  
<input type="text" name="nome" id="nome" />
```

Para quem enxerga, basta visualizar o asterisco para distinguir os campos obrigatórios. Com o leitor de tela, a informação do "alt" seria lida para o usuário. No caso do exemplo de código acima, o leitor iria informar algo do tipo: "Campo de edição, nome, obrigatório".

Outra solução é utilizar uma *tag* "span" contendo a informação "obrigatório", que será ocultada por CSS e, em seu lugar, inserir uma imagem de um asterisco.

```
<label for="nome">Nome:  
  <span>Obrigatório</span>  
</label>  
<input type="text" name="nome" id="nome" />
```

Obs.: Mecanismos para ocultar elementos e seus efeitos na acessibilidade:

Enap
Enap

Enap

Enap
Enap

Mecanismos para ocultar elementos e seus efeitos na acessibilidade.

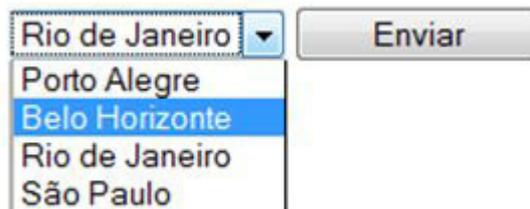
CSS	Efeito na tela	Efeito na acessibilidade
<code>visibility:hidden;</code>	O elemento fica oculto, mas continua a ocupar o espaço que normalmente ocuparia.	O conteúdo é ignorado pelos leitores de tela.
<code>display:none;</code>	O elemento fica oculto e não ocupa espaço.	O conteúdo é ignorado pelos leitores de tela.
<code>height: 0; width: 0; overflow: hidden;</code>	O elemento fica oculto e não ocupa espaço.	O conteúdo é ignorado pelos leitores de tela.
<code>text-indent: -999em;</code>	O conteúdo é movido para "fora da tela", não sendo mais visível, mas <i>links</i> podem ser focalizados de maneira imprevisível.	Os leitores de tela acessam o conteúdo, mas somente texto e elementos <i>inline</i> .
<code>position: absolute; left: -999em;</code>	O conteúdo é removido de sua posição, não ocupando espaço e é movido para "fora da tela", ficando oculto.	Os leitores de tela acessam o conteúdo.

Alterações automáticas

Quando campos ou elementos de um formulário receberem foco, não devem ser iniciadas alterações automáticas na página que confundam ou desorientem os usuários. Logo, as mudanças devem ocorrer por meio do acionamento de botões, permitindo que os usuários tenham total controle do formulário.

Exemplo Correto:

```
<select id="cidade" name="cidade">
  <option value="POA">Porto Alegre</option>
  <option value="BH">Belo Horizonte</option>
  <option value="RJ">Rio de Janeiro</option>
  <option value="SP">São Paulo</option>
</select>
<input type="submit" id="submit" value="Enviar" />
```



CSS:

```
input.btLimpar{
    background:transparenturl(btLimpar.jpg) no-repeat left top;
    width:100px;
    height:47px;
    text-indent:-20000px;
    border:0;
}
```

Aviso de erro

Quando um erro de entrada de dados for automaticamente detectado, seja pelo não preenchimento de campos obrigatórios, ou pelo preenchimento incorreto de algum campo, o item que apresenta erro deve ser identificado e o usuário deve ser informado sobre o erro por meio de texto. Após o envio das informações, caso exista algum campo incorreto, o mesmo deve receber o foco, e nele deverá estar contida uma mensagem informando claramente ao usuário o real motivo do problema, para que este possa ser resolvido, permitindo o envio dos dados.

Exemplo:

O usuário envia o formulário e deixa de preencher campos obrigatórios ou preenche algum campo de maneira incorreta. O foco retorna ao início do formulário contendo o aviso de erro e *links* (âncoras) para os campos do formulário que apresentaram erro. Abaixo dos avisos, encontram-se os campos de formulário que contiveram erro de preenchimento e, também, um *link* que permite ao usuário expandir os demais campos do formulário.

Avisos - dados inválidos

- Usuário inválido, mínimo de 4 caracteres!
- Nome completo inválido, o campo não pode estar em branco!
- Endereço inválido, mínimo de 4 caracteres!
- CEP inválido, somente é permitido a entrada de números!

Dados cadastrais

- Usuário
Mínimo 5 caracteres / sem espaços

- Nome completo

Endereço

- Logradouro

- CEP
Somente números

+ Expandir demais campos

Cadastrar Limpar Cancelar

Após a validação dos dados, antes de enviar o formulário, uma tela de confirmação deverá aparecer, conforme no exemplo a seguir, permitindo que o usuário verifique e, se necessário, edite as informações antes de enviá-las.

Página de Confirmação

Formulário de Contato

Nome: João de Almeida

E-mail: joao.almeida@provedor.com.br

Assunto: Informações

Mensagem: Teste

Editar Enviar

CAPTCHA

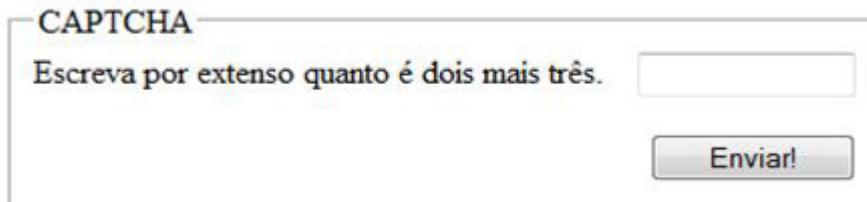
O CAPTCHA é um teste interativo, completamente automatizado, para diferenciar computadores de seres humanos. Os CAPTCHAS comuns fazem uso de imagens que expressam símbolos e caracteres. Esse tipo de CAPTCHA não é interpretado e lido pelos *softwares* leitores de tela. Logo, esse recurso somente deverá ser utilizado em casos de extrema necessidade, e, quando utilizado, deverá ser disponibilizado em forma alternativa, podendo ser de perguntas de interpretação ou testes matemáticos, que só poderão ser respondidas por seres humanos.

Enap
Enap

Essas perguntas devem ser simples, para que possam ser respondidas por variados tipos de usuários, de diferentes culturas e níveis de instrução.

Exemplo:

```
<form action="action.php" method="post">
  <fieldset>
    <legend>CAPTCHA</legend>
    <label for="pergunta">Escreva por extenso quanto é dois mais três.</label>
    <input type="text" id="pergunta" name="pergunta" />
    <input type="submit" name="enviar" value="Enviar!" />
  </fieldset>
</form>
```



2.10 Padrão eGov

Enap

Esta seção demonstra os elementos padronizados de acessibilidade digital que devem estar presentes em todos os sítios do governo federal, para facilitar o acesso de todos.

Esses elementos deverão estar disponibilizados na chamada Barra de Acessibilidade, que deve encontrar-se no topo de todas as páginas:



Enap
Enap

```
<div id="acessibilidade">
  <ul id="atalhos">
    <li><a href="#iniciodoconteudo">Ir para o conteúdo [1]</a></li>
    <li><a href="#iniciodomenu">Ir para o menu [2]</a></li>
    <li><a href="#busca">Ir para a busca [3]</a></li>
    <li><a href="#iniciodorodape">Ir para o rodapé [4]</a></li>
  </ul>
  <ul id="botoes">
    <li><a href="acessibilidade.html">Acessibilidade </a></li>
    <li><a href="#" id="bt_contraste">Alto contraste</a></li>
    <li><a href="mapa.html"> Mapa do sítio </a></li>
  </ul>
</div>
```

Esses elementos incluem:

Atalhos

Os atalhos que deverão existir nas páginas do governo federal são os seguintes:

- 1: para ir ao conteúdo;
- 2: para ir ao menu principal;
- 3: para ir à caixa de pesquisa.

Ir para o conteúdo 1 Ir para o menu 2 Ir para a busca 3

Atalhos para outros pontos, como para o rodapé, poderão ser disponibilizados também, mas os três atalhos citados anteriormente são obrigatórios nos sítios governamentais, seguindo-se o padrão da numeração.

As dicas dos atalhos deverão ser disponibilizadas na barra de acessibilidade e na página sobre a acessibilidade do sítio.

Alto contraste

A opção de alto contraste deve gerar uma página em que a relação de contraste entre o plano de fundo e os elementos do primeiro plano seja de, no mínimo 7:1 (contraste otimizado). Dessa forma, a folha principal de alto contraste deve obedecer à seguinte configuração de cores:

- **Cor de fundo:** independentemente da cor utilizada, ela deve ser alterada para preto (#000000);
- **Cor de texto:** independentemente da cor utilizada, ela deve ser alterada para branco (#FFFFFF);
- **Links:** o modo normal do *link* deve ser sublinhado (para que ele se diferencie do texto normal), assim como o modo "hover" e o modo "active". O *link* deve ser alterado para amarelo (#FFF333);

Enap
Enap

Enap

Enap
Enap

Exemplo de uma parte do mapa do sítio:



Enap
Enap

Enap

Enap
Enap

No exemplo acima, todos os itens são *links* que remetem as suas páginas correspondentes.

É importante salientar que em sítios muito simples, em que os *links* para as páginas já estão todos presentes no menu, não existe a necessidade de mapa do sítio.



Saiba Mais sobre a identidade digital do governo

Esses elementos já fazem parte da identidade digital de governo. Visite o sítio [Portal Padrão \(link para um novo sítio\)](#) para mais detalhes dessa iniciativa.



3 Avaliação de Acessibilidade

A avaliação de acessibilidade é um processo fundamental para garantir uma Web acessível. Existem ferramentas automáticas que auxiliam na avaliação de acessibilidade, como o [Avaliador e Simulador de Acessibilidade de Sítios \(ASES\) \(link para um novo sítio\)](#), capaz de avaliar os itens do eMAG que são passíveis de validação por máquina.

Apesar de muito úteis, essas ferramentas sozinhas não são capazes de detectar todos os problemas de acessibilidade de um sítio. Para tal, faz-se análise manual, que pode ser realizada por usuários reais, utilizando diferentes tipos de tecnologia, e por especialistas na área de acessibilidade Web.

O eMAG sugere um passo a passo para avaliar a acessibilidade de um sítio:

1. Validar os códigos de conteúdo HTML e folhas de estilo;
2. Verificar o fluxo de leitura da página, utilizando um navegador textual ou um *software* leitor de tela;
3. Checar o fluxo de leitura das páginas sem estilos, sem *scripts* e sem as imagens;
4. Verificar as funcionalidades disponíveis na barra de acessibilidade, como atalhos e contraste;
5. Efetuar a validação automática de acessibilidade por meio de avaliadores automáticos;
6. Realizar a validação manual, com uso de *checklists* de validação humana.

Um recurso bastante útil na avaliação de acessibilidade de um sítio são as extensões para navegadores para análise de acessibilidade:

- Para Firefox: [Accessibility Evaluation Toolbar \(link para um novo sítio\)](#).
- Para Google Chrome: [Accessibility Developer Tools \(link para um novo sítio\)](#).
- Para Internet Explorer: [Web Accessibility Toolbar for Internet Explorer \(link para um novo sítio\)](#).



A avaliação manual pode ser feita utilizando-se *Checklists*, como as disponibilizadas pelo Departamento de Governo Digital:

- Checklist Manual de Acessibilidade - Desenvolvedores.
- Checklist Manual de Acessibilidade - Usuários com deficiência visual.
- Ou utilizando o anexo D - Pontos de Checagem validáveis por avaliação Humana (.ods, 26kb) [\(link para outro site\)](#) da Cartilha de Contratação [\(link para um novo site\)](#).



Saiba Mais sobre avaliação de acessibilidade

- [Lista completa de ferramentas para avaliação de acessibilidade - em inglês \(link para um novo sítio\).](#)



4 Recursos e ferramentas

Organizações e especificações

- [World Wide Web Consortium \(W3C\) \(link para um novo sítio\)](#)
- [Web Content Accessibility Guidelines \(WCAG\) Versão 2.0 \(link para um novo sítio\)](#)
- [Web Content Accessibility Guidelines \(WCAG\) Versão 2.0 \(português\) \(link para um novo sítio\)](#)
- [Modelo de Acessibilidade em Governo Eletrônico \(eMAG\) \(link para um novo sítio\)](#)

Validadores de código

- [Validador de Extensible Hypertext Markup Language \(XHTML\) e Hypertext Markup Language \(HTML\) \(link para um novo sítio\)](#)
- [Validador de Folhas de Estilo \(CSS\) \(link para um novo sítio\)](#)
- [Validador de links \(link para um novo sítio\)](#)

Validadores automáticos de acessibilidade

- [Avaliador e Simulador de Acessibilidade de Sítios \(ASES\) \(WAI e eMAG\) \(link para um novo sítio\) Da Silva \(WAI e eMAG\) \(link para um novo sítio\)](#)
- [AccessMonitor \(WAI\) \(link para um novo sítio\)](#)
- [Web Accessibility Tool \(WAVE\) \(WAI\) \(link para um novo sítio\)](#)
- [Cynthia Says \(WAI\) \(link para um novo sítio\)](#)
- [Hera \(WAI\) \(link para um novo sítio\)](#)
- [Lift \(WAI\) \(link para um novo sítio\)](#)
- [TAW \(WAI\) \(link para um novo sítio\)](#)
- [Examinator \(WAI\) \(link para um novo sítio\)](#)
- [Functional Accessibility Evaluator 2.0 \(WAI e Section 508\) \(link para um novo sítio\)](#)

Barra de ferramentas para avaliação de acessibilidade

- [Web Accessibility Toolbar for Internet Explorer \(para Internet Explorer\) \(link para um novo sítio\)](#)
- [Accessibility Evaluation Toolbar \(para Firefox\) \(link para um novo sítio\)](#)

Ferramentas para análise de relação de contraste

- [Luminosity Colour Contrast Ratio Analyser \(online\) \(link para um novo sítio\)](#)
- [Contrast Analyser \(online\) \(link para um novo sítio\)](#)
- [Check my colours \(online\) \(link para um novo sítio\)](#)
- [Contrast Analyser 2.2 \(link para um novo sítio\)](#)

Simulador de Navegador

- [Browsershots \(link para um novo sítio\).](#)

Enap
Enap

Enap

Enap
Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Enap

Navegadores Textuais

[Lynx \(link para um novo sítio\).](#)

[LynxViewer \(simulador\) \(link para um novo sítio\).](#)

Simulador para Cegueira Cromática (Daltonismo)

[Vischeck \(link para um novo sítio\).](#)

Leitores de Tela

[Jaws for Windows - Leitor de tela americano produzido pela Freedom Scientific \(link para um novo sítio\)](#)

[NVDA - Leitor de tela gratuito e de código aberto para Windows \(link para um novo sítio\)](#)

[Orca - Leitor de tela gratuito e de código aberto para Linux \(link para um novo sítio\)](#)

[Virtual Vision - Leitor de tela nacional fabricado pela Micropower \(link para um novo sítio\)](#)

[VoiceOver - Leitor de tela para MAC OS \(link para um novo sítio\)](#)

[Windows Eyes - Leitor de tela canadense fabricado pela GW Micro \(link para um novo sítio\)](#)

[DOSVOX - Interface especializada desenvolvida pela UFRJ \(link para um novo sítio\)](#)

Material de apoio

[Material de apoio fornecido pelo Departamento de Governo Digital, incluindo tutoriais de acessibilidade e modelos de checklists \(link para um novo sítio\)](#)

[Cartilha de acessibilidade na Web do W3C \(link para um novo sítio\)](#)

[Lista completa de ferramentas para avaliação de acessibilidade disponibilizada pelo W3C \(em inglês\) \(link para um novo sítio\)](#)

[Building Accessible Websítios \(em inglês\) - livro *online* bastante completo sobre como desenvolver um sítio acessível \(link para um novo sítio\)](#)

5 Glossário

Atributo

Os atributos servem para definir uma propriedade de um elemento (X)HTML.

Folhas de estilo ou CSS (Cascading Style Sheets)

Representam uma linguagem de estilização, ou seja, utilizada para definir a apresentação visual de uma página. Assim, enquanto o conteúdo de um sítio encontra-se no HTML, a formatação é feita por meio das folhas de estilo, separando-se, dessa forma, as camadas lógicas.

HTML (Hyper Text Markup Language)

É uma linguagem de marcação utilizada para produzir páginas da Web.

Leitor de tela

São *softwares* que interagem com o sistema operacional, capturando as informações apresentadas na forma de texto e transformando-as em resposta falada por meio de um

