



Departamento de Governo Eletrônico

Secretaria de Logística e Tecnologia da Informação
Ministério do Planejamento, Orçamento e Gestão

www.governoeletronico.gov.br

Recomendações de Acessibilidade para a Construção e Adaptação de Conteúdos do Governo Brasileiro na Internet

eMAG, Acessibilidade de Governo Eletrônico **Cartilha Técnica**

Documento de Referência
Versão 2.0
14 de Dezembro de 2005

SUMÁRIO

1	Introdução	3
2	Recomendações de Acessibilidade para a Construção e Adaptação de Conteúdos na Internet.....	4
2.1	Visão Técnica do Modelo Proposto.....	4
2.2	Diretrizes Técnicas de Acessibilidade	4
3	Níveis de Acessibilidade e Recomendações	6
3.1	Nível de Prioridade 1	6
3.1.1	<i>Casos gerais.....</i>	6
3.1.2	<i>Casos em que são utilizados imagens e mapas de imagem</i>	10
3.1.3	<i>Casos em que são utilizadas tabelas</i>	13
3.1.4	<i>Casos em que são utilizados frames.....</i>	15
3.1.5	<i>Casos em que são utilizados applets, objects, scripts e programas interpretáveis</i>	16
3.1.6	<i>Casos em que é utilizada multimídia.....</i>	17
3.2	Nível de Prioridade 2	19
3.2.1	<i>Casos gerais.....</i>	19
3.2.2	<i>No caso de serem utilizadas tabelas.....</i>	26
3.2.3	<i>No caso de serem utilizados formulários.....</i>	27
3.2.4	<i>No caso de serem utilizados applets e programas interpretáveis</i>	29
3.3	Nível de Prioridade 3	32
3.3.1	<i>Casos Gerais.....</i>	32
4	Relação entre as Áreas de Acessibilidade compreendidas na Visão do Cidadão e as Recomendações da Visão Técnica	40
5	Detalhes da Avaliação e Validação.....	42
5.1	Programas Avaliadores de Acessibilidade	42
5.2	Programas Leitores de Tela	43
6	Leitura complementar.....	44

1 Introdução

Buscando atender e propiciar a acessibilidade dos sítios governamentais, como proposto em “eMAG, Acessibilidade de Governo Eletrônico – Modelo”, foi desenvolvida esta Cartilha Técnica de recomendações. Enquanto a Visão do Cidadão – tratada no documento Modelo de Acessibilidade – preocupa-se com o entendimento facilitado do Modelo de Acessibilidade para a maioria dos cidadãos brasileiros, a Visão Técnica focaliza o trabalho em uma forma adequada para um outro perfil de pessoa, o desenvolvedor de sítios e profissionais relacionados a esta tarefa.

O presente documento propõe-se a detalhar a Visão Técnica do Modelo de Acessibilidade, expressando detalhadamente nossa proposta para a implementação das Recomendações de Acessibilidade para a Construção ou Adaptação de Conteúdos do Governo Brasileiro na Internet.

2 Recomendações de Acessibilidade para a Construção e Adaptação de Conteúdos na Internet

2.1 Visão Técnica do Modelo Proposto

Esta cartilha está dividida, de forma macro, conforme os Níveis de Prioridade de Acessibilidade, descritos na Visão do Cidadão do modelo de acessibilidade proposto – documento “eMAG, Acessibilidade de Governo Eletrônico - Modelo”.

São 3 níveis de prioridade de acessibilidade, sendo o *Nível Prioridade de Acessibilidade 1* as exigências básicas de acessibilidade; o *Nível Prioridade de Acessibilidade 2* com as normas e recomendações que sendo implementadas garantem o acesso às informações do documento; e o *Nível Prioridade de Acessibilidade 3* com normas e recomendações que sendo implementadas facilitarão o acesso aos documentos armazenados na Web.

Dentro de cada nível, as Recomendações de Acessibilidade subdividem-se de acordo com suas características técnicas, facilitando mais ainda o processo de operacionalização da acessibilidade dos conteúdos.

Além destes dois agrupamentos, a Visão Técnica do modelo define uma camada de abstração intermediária entre suas recomendações e a Visão do Cidadão, que é chamada de “Diretrizes Técnicas de Acessibilidade de Governo Eletrônico”, assunto que será abordado a seguir.

2.2 Diretrizes Técnicas de Acessibilidade

No item 2.1 desta cartilha foi descrita a forma de organização da presente cartilha, sendo as recomendações para acessibilidade de conteúdos divididas em níveis de prioridade e, dentro de cada nível, pelo tipo de envolvimento técnico necessário para implementar estas recomendações. Estas recomendações serão tratadas no item 3.

As Diretrizes Técnicas de Acessibilidade são uma outra forma de visualizar e compreender as Recomendações de Acessibilidade, não baseadas em suas prioridades de implementação,

mas na facilidade do processo cognitivo e de aprendizado dos técnicos envolvidos neste assunto. A diretriz se preocupa com o impacto técnico, o resultado da implementação das recomendações.

As Diretrizes Técnicas de Acessibilidade de Governo Eletrônico buscam facilitar ao técnico a compreensão, a fixação e o domínio das recomendações aqui propostas. Elas não definem uma ordem de implementação, que cabe ao modelo de Níveis de Acessibilidade, contudo, agrupam as recomendações de acordo com a percepção do resultado.

Diretriz 1. Forneça alternativas equivalentes para o conteúdo gráfico e sonoro.

Recomendações: 1.10, 1.11, 1.20, 1.21.

Diretriz 2. Assegure-se de que seu sítio seja legível e compreensível mesmo sem o uso de formatações.

Recomendações: 1.4, 1.5, 1.6, 1.13, 1.14, 1.15, 2.12, 2.13, 3.2.

Diretriz 3. Dê preferência às tecnologias de marcação e formatação.

Recomendações: 2.1, 2.2, 2.3, 2.7, 2.8, 2.10.

Diretriz 4. Assegure que toda a informação seja interpretada corretamente, com clareza e simplicidade.

Recomendações: 1.1, 1.2, 1.9, 1.18, 1.19, 2.9, 2.11, 2.14, 2.15, 3.2, 3.5, 3.11.

Diretriz 5. Assegure que as tecnologias utilizadas funcionem - de maneira acessível - independente de programas, versões e futuras mudanças.

Recomendações: 1.16, 1.17, 1.19, 1.20.

Diretriz 6. Assegure sempre o controle do usuário sobre a navegação no sítio.

Recomendações: 1.7, 1.8, 1.23, 2.4, 2.5, 2.6, 2.19, 3.1, 3.3.

Diretriz 7. Identifique claramente quais são os mecanismos de navegação.

Recomendações: 1.3, 1.10, 2.16, 3.6, 3.7, 3.8, 3.9, 3.10, 3.12, 3.13.

Diretriz 8. Em casos não contemplados pelas diretrizes anteriores, utilize sempre recursos reconhecidos, por instituições com propriedade no assunto, como **tecnologias acessíveis**.

Recomendação: 1.24.

3 Níveis de Acessibilidade e Recomendações

Conforme o documento da Visão do Cidadão do Modelo de Acessibilidade proposto “eMAG, Acessibilidade de Governo Eletrônico – Modelo”, é a clara necessidade de ser proposta uma acessibilidade evolutiva, baseada em prioridades.

Os níveis de prioridades não somente agrupam as recomendações conforme suas prioridades de implementação, como também agrupam as recomendações conforme suas características técnicas de implementação.

Propõe-se que este modelo de níveis de prioridades seja seguido, implementando-se primeiro as recomendações do nível de prioridade 1, seguidas pelas recomendações do nível de prioridade 2 e, finalmente, pelas recomendações do nível de prioridade 3, conforme descrito abaixo.

3.1 Nível de Prioridade 1

As recomendações de Nível de Prioridade 1 referem-se às exigências básicas de acessibilidade. Pontos em que os criadores e adaptadores de conteúdo Web devem satisfazer inteiramente. Se não cumpridas, grupos de usuários ficarão impossibilitados de acessar as informações do documento.

3.1.1 Casos gerais

Recomendação 1.1 - Identificar o principal idioma utilizado nos documentos. O idioma do documento deve ser especificado na expressão HTML. Neste exemplo, o idioma principal do conteúdo é italiano.

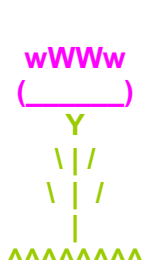
```
<html lang="pt-br">
<head></head>
<body><p>
Gostaria de saber a origem da feijoada? Muitos atribuem aos escravos, outros aos
portugueses, outros ainda aos franceses.
</p></body></html>
```

Recomendação 1.2 - Identificar claramente quaisquer mudanças de idioma no texto de um documento, bem como nos equivalentes textuais (por ex., legendas de imagens). Use o atributo "lang" para identificar claramente as alterações do idioma no texto.

```
<html lang="pt-br">
<head></head>
<body><p>
Mãe, ele está dizendo para ir. Ele disse, <span lang="fr"> "Allons, Madame
plaisante!"
</p></body></html>
```

Recomendação 1.3 - Fornecer meios para ignorar e explicar inserções de arte ASCII com várias linhas, tais como *links* de âncora ou páginas alternativas.

Exemplo: Abaixo um exemplo de código que pode ser usado para saltar arte ASCII :



```
<p>
<a href="#post-art">saltar por cima do arte ASCII</a>
<!-- Arte ASCII entra aqui -->
<a name="post-art">Primeira linha de texto do corpo principal ...</a>
```

Recomendação 1.4 - Assegurar que todas as informações veiculadas com cor estejam também disponíveis sem cor.

Existem dois procedimentos para executar a tarefa, o mais indicado é o procedimento em vermelho.

- Procedimento azul.
- Procedimento vermelho.

Nesse exemplo, para os casos em que o usuário não possa ver através de um monitor colorido, é mais fácil localizar a informação porque está localizada textualmente.

Recomendação 1.5 - Assegurar que a combinação de cores entre o fundo e o primeiro plano seja suficientemente contrastante para poder ser vista por pessoas com cromodeficiências, bem como pelas que utilizam monitores de vídeo monocromáticos.

Usar uma combinação de cores que ofereça um bom contraste entre texto e o fundo. Produza para suas páginas um conjunto de cores que possa ser visto tanto em monitores coloridos como em monitores monocromáticos.



Recomendação 1.6 - Organizar os documentos de tal forma que possam ser lidos sem recurso à folhas de estilo. Por exemplo, se um documento em HTML for reproduzido sem as folhas de estilo que lhe estão associadas, deve continuar a ser possível lê-lo.

Recomendação 1.7 - Não usar concepções que possam provocar intermitência da tela, até que os leitores de tela ou navegadores do usuário possibilitem o seu controle.

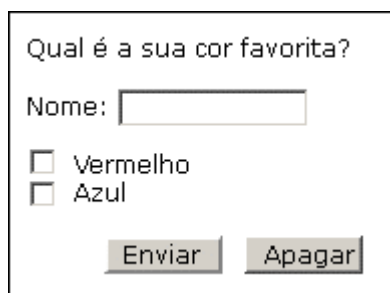
Exemplo: Podem ocorrer no uso de *flash*, *applet*, gif animado, *scripts*, entre outros, que permitem imagens piscantes.

Obs: Pessoas com epilepsia fotosensitiva podem desencadear um ataque epilético com o cintilar ou piscar numa banda de 4 a 59 intermitências por segundo (Hertz), com um pico nas

20 intermitências por segundo, bem como com alterações rápidas do escuro para a luz (como sucede com as lâmpadas das discotecas).

Recomendação 1.8 - Criar uma seqüência lógica de tabulação para percorrer *links*, controles de formulários e objetos.

A tecla TAB pode ser usada para navegar dentro das páginas. Portando, as marcações devem ser dispostas para que sejam navegadas de forma linear. O comando TABINDEX permite que você estabeleça uma ordem lógica, ao usar a tecla TAB eles surgem na ordem “correta” (para navegadores que suportam o atributo “tabkey”). O TABINDEX trabalha com os elementos: <a>, <area>, <button>, <input>, <object>, <select> e <textarea>.



No código html:

```
<form method="post" action="home.htm">  
Qual é a sua cor favorita? <br> <br>  
Nome: <input type="text" name="tboxin" tabindex="1" size="20">  
<br><br>  
<input type="checkbox" name="cb1" value="checkbox1" tabindex="2"> Vermelho <br>  
<input type="checkbox" name="cb3" value="checkbox3" tabindex="3"> Azul <br>  
<input type="submit" value="enviar" tabindex="4">  
<input type="reset" value="apagar" tabindex="5">  
</form>
```

Apesar de poder controlar a ordem da tecla TAB, o ideal é que você crie os formulários de forma lógica. O TABINDEX é altamente recomendado para o caso dos campos obrigatórios de formulários extensos.

Recomendação 1.9 - Utilizar a linguagem mais clara e simples possível, logicamente, adequada ao conteúdo do sítio. Siga estas instruções de redação:

- Crie cabeçalhos e as descrições dos *links* de forma clara e concisa. Avalie todos os cabeçalhos, notas, e menus para ver se as palavras chaves significam exatamente o que se pretende, e se existem mais palavras em comum que tenham o mesmo significado;
- Coloque o tópico da frase ou parágrafo no início de um desses elementos;

- Limite cada parágrafo a uma idéia principal;
- Evite linguagem idiomática, jargão técnico, expressões e vocabulário não familiar;
- Evite o uso de linguagem especializada com vocabulário familiar, a não ser que seja fornecida uma explicação mais extensa;
- Evite a voz passiva;
- Evite uma estrutura de frases complexa;
- Construa frases curtas para os *links*, mas com suficiente significado, de forma que façam sentido quando lidas fora do contexto, isoladas ou como parte de uma série de *links*.

Recomendação 1.10 - Utilizar os mecanismos de navegação de maneira coerente, consistente e sistemática. Por exemplo, organizar itens do menu por tema, seções ou classes, etc.

A padronização do *layout*, ou seja, um mesmo estilo de apresentação em cada página, permite aos usuários encontrar facilmente os botões de navegação assim como procurar o conteúdo anterior em cada página. Enquanto isto ordena logicamente, resulta em grande benefício para pessoas com incapacidade de leitura e aprendizagem. Torna também previsível a localização da informação necessária em cada uma das páginas, maximizando desta forma sua localização.

3.1.2 Casos em que são utilizados imagens e mapas de imagem

Recomendação 1.11 - Fornecer um equivalente textual a cada imagem (isso abrange: representações gráficas do texto, incluindo símbolos, GIFs animados, imagens utilizadas como sinalizadores de pontos de enumeração, espaçadores e botões gráficos), para tanto, utiliza-se o atributo “alt” ou “longdesc” em cada imagem.

Imagens decorativas ou de acabamento: no caso de uma imagem ter apenas um fim decorativo ou de acabamento, como o canto arredondado de uma tabela ou uma régua separadora utiliza-se o atributo alt com um espaço em branco.

```

```

Para uma imagem meramente ilustrativa ou representações gráficas de texto, faz-se apenas uma breve descrição desta:



No código html:

```

```

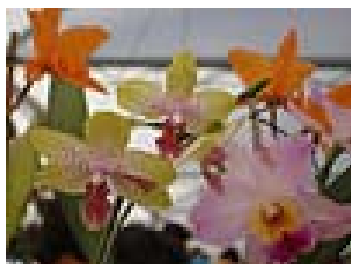
A
e o Pedra
Metal

No código html:

```

```

O atributo "longdesc" foi criado para situações em que a descrição a ser feita deve ser mais longa do que a permitida pelo atributo "alt".



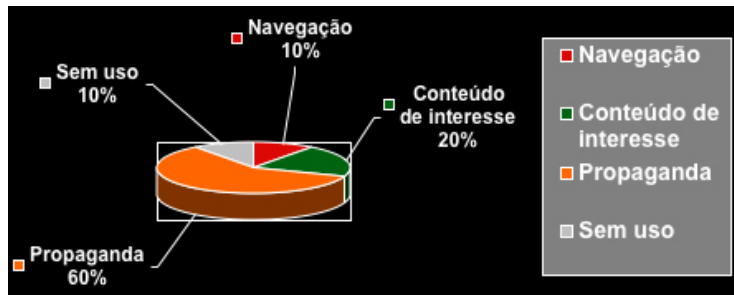
No código html:

```

```

O documento "orquideas.htm" possui um texto com a descrição mais detalhada da imagem.

No caso de gráficos assegurar que os leitores de tela possam extrair a informação significativa. Por exemplo, uma descrição textual da informação transmitida através de um gráfico em pizza deve estar disponível através da utilização do "alt" ou "longdesc" de maneira que os dados do gráfico possam ser compreendidos pelo usuário.



No código html:

```

```

Recomendação 1.12 - Fornecer *links* de texto redundantes relativos a cada região ativa de um mapa de imagem armazenado tanto no cliente quanto no servidor. Não esquecendo de adicionar texto equivalente à imagem mostrada, no caso o “alt” ou “longdesc”.

No código html:



```

<map name="map1">
<area coords="0,0,39,39" href="a.htm" alt="link seção a do sítio. ">
<area coords="40,0,79,39" href="b.htm" alt="link seção b do sítio. ">
</map> <map name="map2">
[<a href="a.htm">seção a</a> | <a href="b.htm">seção b</a> ]
</map>
```

Observação: o texto “alt” que se encontra no elemento “img” informa ao usuário que existe um equivalente textual. Não descreve, no entanto, a imagem por si só, para tanto, utiliza-se o atributo “longdesc”.

3.1.3 Casos em que são utilizadas tabelas

Recomendação 1.13 - Fornecer resumos das tabelas utilizando o atributo “summary”, caso seja criada uma tabela para dados. Se a tabela foi criada para efeito de *design*, deixar o “summary” em branco.

```
<table border="1" summary="Esta tabela contém os dados relativos ao número de
xícaras de café">
...código da tabela...
</table>
```

Recomendação 1.14 - Em tabelas de dados com dois ou mais níveis lógicos de cabeçalhos, sejam de linha ou de coluna, utilizar marcações para associar as células de dados às células de cabeçalho. Organize tabelas complexas de forma que possa identificar facilmente suas divisões.

A tabela seria mostrada da seguinte maneira no navegador:

Despesas de Viagem (custo actual, euros)

VIAGEM, data	Refeições	Alojamento	Transporte	Total
Lisboa				
25 Ago 97	37.74	112.00	45.00	
26 Ago 97	27.28	112.00	45.00	
Subtotal	65.02	224.00	90.00	379.02
Porto				
27 Ago 97	96.25	109.00	36.00	
28 Ago 97	35.00	109.00	36.00	
Subtotal	131.25	218.00	72.00	421.25

A seguir o código a ser utilizado:

```

<table border="1" cellpadding=2 cellspacing=3>
<caption>Despesas de Viagem (custo actual, euros)</caption>
<thead> <tr>
<th><p><span id="t-l1">VIAGEM</span></p><br>
<span id="t-l2"> data</span></p></th>
<th scope="column">Refeições</th>
<th scope="column">Alojamento</th>
<th scope="column"><abbr="Transporte">Trans.</abbr></th>
<th scope="column">Total</th>
</tr> </thead>
<tbody>
<tr> <th scope="rowgroup" headers="t-l1">Lisboa</th>
</tr> <tr>
<td scope="row" headers="t-l2"> 25 Ago 97</td>
<td>37.74</td> <td>112.00</td> <td>45.00</td>
</tr> <tr>
<td scope="row" headers="t-l2"> 26 Ago 97</td>
<td>27.28</td> <td>112.00</td> <td>45.00</td>
</tr> <tr>
<td scope="row">Subtotal</td>
<td>65.02</td> <td>224.00</td> <td>90.00</td> <td>379.02</td>
</tr> </tbody> <tbody> <tr>
<th scope="rowgroup" headers="t-l1">Porto</th>
</tr> <tr>
<td scope="row" headers="t-l2"> 27 Ago 97</td>
<td>96.25</td> <td>109.00</td> <td>36.00</td>
</tr> <tr>
<td scope="row" headers="t-l2"> 28 Ago 97</td>
<td>35.00</td> <td>109.00</td> <td>36.00</td>
</tr> <tr>
<td scope="row">Subtotal</td>
<td>131.25</td> <td>218.00</td> <td>72.00</td> <td>421.25</td>
</tr> </tbody> </table>

```

Nota: no caso de utilizar tabelas para formatar páginas (em vez de CSS), então NÃO deve usar a notação existente para tabelas de dados - como "th", "thead", "tbody", "scope", "colgroup", etc - pois são elementos utilizados para identificar e manipular os dados.

3.1.4 Casos em que são utilizados frames

Recomendação 1.15 - Assegurar que os equivalentes de conteúdo dos *frames* (dinâmico ou não) sejam atualizados sempre que esse conteúdo mudar. A origem do frame sempre deve estar ligada a um arquivo HTML.

Forma correta

```
<frame name="frame_foto" src="home.html">
```

Em que "home.html" deve conter o seguinte:

```

```

Forma errada

```
<frame name="frame_foto" src="home.gif">
```

Se a fonte do *frame* for próprio arquivo de imagem, não é possível dar um equivalente textual à imagem. A descrição do frame cabe ao documento HTML chamado pelo frame.

Recomendação 1.16 - Assegurar a acessibilidade do conteúdo de *frames*, fornecendo uma página alternativa através do elemento "noframes".

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Frameset//EN">
<html lang="pt-br">
<head>
  <title>este é o topo.html</title> </head>
  <frameset cols="50%, 50%" title="O nosso documento grande">
    <frame src="main.html" title="Onde os assuntos do índice são visualizados">
    <frame src="table_of_contents.html" title="Índice"> </frameset> <noframes>
  <body> <a href="table_of_contents.html">Índice.</a>
  <!-- os outros links de navegação que se encontram disponíveis em main.html encontram-se
  aqui também. -->
</body> </noframes> </html>
```

O código abaixo cria um *layout* com dois *frames* em navegadores que suportam *frames*, enquanto que o elemento "noframes" gera uma única janela de conteúdo que é mostrada nos navegadores que não suportam *frames*.

Recomendação 1.17- Dar a cada *frame* um título que facilite a identificação dos *frames* e sua navegação.

```
<frame src="f1.htm" title="Título e frame da barra de navegação principal">  
<frame src="f3.htm" title="Frame para visualização do conteúdo">
```

Observação: O mesmo processo ocorre com “<iframes>”.

Recomendação 1.18 - Descrever a finalidade dos *frames* e o modo como se relacionam entre si, se isso não for óbvio a partir unicamente dos títulos, forneça uma descrição mais detalhada.

```
<frame src="main.htm" longdesc="maindesc.htm" title="Frame do Conteúdo Principal.">
```

Caso seja necessário, explicar com mais detalhes sobre a finalidade de cada *frame*, use o atributo “longdesc” para complementar a informação.

3.1.5 Casos em que são utilizados *applets*, *objects*, *scripts* e programas interpretáveis

Recomendação 1.19 - Assegure a acessibilidade de objetos programados, tais como programas interpretáveis e *applets*, garantindo que a resposta a eventos seja independente do dispositivo de entrada e que qualquer elemento dotado de interface própria possa funcionar com qualquer leitor de tela ou navegador que o usuário utilize. Evite colocar *scripts* que estejam vinculados a *links*, se isso não for possível, fornecer informações equivalentes em uma página alternativa acessível.

NÃO use: Cadastre-se agora!

Evite a criação de *links* que usem “javascript” tais como URL. Se um usuário não usar *scripts*, então não será capaz de encontrar os *links*, uma vez que o navegador não consegue criar o conteúdo *link*. Isto é um *link* considerado como “beco sem saída” para um navegador em que os *scripts* não são suportados ou gravados.

Recomendação 1.20 - Assegurar que todas as páginas possam ser utilizadas mesmo que os programas interpretáveis, os *applets* ou outros objetos programados tenham sido desativados ou não sejam suportados. Sempre que tiver script associe logo a seguir o elemento “noscript”.

Aqui está o código usado para criar um botão “scripted”:

```
<a href="javascript: pop()">Cadastre-se agora!</a>
```

E o código da função *script* propriamente:

```
<script language="JavaScript" type="text/javascript">
```

```
function pop() { alert ("Você Ganhou o Primeiro Prémio!") }
```

```
</script>
```

```
<noscript>
```

```
Você Ganhou o Primeiro Prémio! Siga para o cadastro do sítio.
```

```
</noscript>
```

3.1.6 Casos em que é utilizada multimídia

Recomendação 1.21 - Fornecer equivalentes textuais para sons (reproduzidos ou não com interação do usuário), arquivos de áudio independentes, trilhas áudio de vídeo e trechos de vídeo.

Se na aplicação é utilizado um *script* (ou um programa) que causa um aviso sonoro, para ser tocado se o visitante da página tentar enviar um formulário antes dos campos requeridos estarem preenchidos, deve-se incluir no programa, ou *script*, a capacidade de passar, pro escrito, uma mensagem na tela, tal como:

```
Alerta: "Você tentou submeter um formulário incompleto.  
Por favor, preencha os campos necessários".
```

Recomendação 1.22 - Em apresentações multimídia baseadas em tempo (filme ou animação), fornecer ou sincronizar alternativas textuais equivalentes (legendas ou descrições sonoras dos trechos visuais).

Uma apresentação pode incluir qualquer forma de multimídia, como um filme, animação ou apresentação com *slides*. As legendas (que fornecem acesso às faixas áudio) e as descrições em áudio (que fornecem acesso às faixas visuais) constituem alternativas equivalentes a estes tipos de apresentação. A faixa que contém a legenda é uma alternativa para espectadores surdos ou com dificuldades auditivas. A faixa de descrição em áudio é uma alternativa para pessoas cegas ou com baixa visão.

Mantenha sempre presente que para pessoas que não têm acesso aos dispositivos de reprodução multimídia, ou para pessoas surdas ou cegas, a transcrição do áudio e do áudio-descrição continuam a ser as melhores alternativas.

Recomendação 1.23 - Evitar páginas contendo movimento, até que os agentes do usuário possibilitem o controle e a imobilização do conteúdo.

Quando uma página incluir conteúdo em movimento, forneça um mecanismo dentro do script ou do *applet* que permita aos usuários “congelar” o movimento ou atualizá-lo, quando o usuário desejar.

Ao usar folhas de estilo com *script* para criar movimento, dê a possibilidade ao usuário de desligar ou alterar os parâmetros de uma forma fácil.

Recomendação 1.24 – Não sendo possível criar uma página acessível, crie uma página alternativa, juntamente com uma justificativa apropriada, que utilize tecnologias em conformidade com este documento - acessível, que contenha informações (ou funcionalidade) equivalentes e seja atualizada tão frequentemente quanto a página original, considerada inacessível.

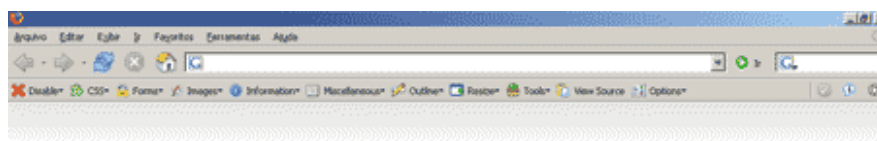
<H1>Bem-vindo ao Sítio Web da Organização!</H1>

Siga este link se pretende
 um sítio sem a preocupação com
acessibilidade

 ou siga este link se pretende

uma versão acessível.

Veja como o código fica no navegador:



Bem-vindo ao Sítio Web da Organização!

Siga este link se pretende:

[um sítio sem a preocupação com a acessibilidade](#)

ou siga este link se pretende:

[uma versão acessível](#)



3.2 Nível de Prioridade 2

Normas e Recomendações de Acessibilidade cuja implementação garante o acesso às informações do documento. Se não cumpridas, grupos de usuários terão dificuldades para navegar e acessar as informações do documento.

3.2.1 Casos gerais

Recomendação 2.1 - Criar documentos passíveis de validação por gramáticas formais publicadas. Declarando o tipo de documento (atributo "doctype") no topo do código fonte de cada página do sítio. Assim seu sítio informará aos servidores, navegadores e validadores que

o código está dentro das regras da linguagem utilizada. Para maiores informações acesse a página do W3 Consortium sobre o assunto: <http://www.w3.org/QA/Tips/Doctype>.

Recomendação 2.2 - Utilizar unidades relativas, e não absolutas, nos valores dos atributos de tabelas, textos, etc. Em CSS não use valores absolutos como “pt” ou “px” e sim valores relativos como o “em”, “ex” ou em porcentagem.

```
Ao invés de usar: p {font-size: 14pt;}  
Use: p {font-size: 1em;}  
Ou ainda: p {font-size: 120%;}
```

Recomendação 2.3 - Marcar corretamente listas e pontos de enumeração em listas ordenadas. Use corretamente a estrutura e os itens das listas. Evite o uso destes elementos para formatar parágrafos.

- A. Lista de instrumentos encontrados numa banda de rock:
1. Guitarra
 2. Baixo
 3. Bateria

No código html:

```
<ol type=a> <li>Lista de instrumentos  
encontrados numa banda de rock:</li>  
<ol type=1>  
<li>Guitarra</li>  
<li>Baixo</li>  
<li>Bateria</li>  
</ol> </ol>
```

Recomendação 2.4 - Não criar páginas com atualização automática periódica, até que os leitores de tela ou navegadores possibilitem o controle da atualização para o usuário. Não utilize a tag meta “refresh” ou dispositivos semelhantes para atualização da página. Caso a página seja continuamente atualizada, informe ao usuário que ele deve “recarregar” a página de tempos em tempos.

```
Exemplo do que não deve ser usado:  
<meta http-equiv="refresh" content="60"> <body> <p>...informação... </body>
```

Este código força a atualização da página a cada 60 segundos.

Em páginas que possuem limite de tempo para serem respondidas, o usuário deve ser alertado antes do término do tempo, e deverá ser dado um tempo suficiente para leitura e preenchimento.

Recomendação 2.5 - Não utilizar marcações para redirecionar as páginas automaticamente, até que os leitores de tela ou navegadores do usuário possibilitem interromper o processo.

Nunca use os atributos meta-refresh para redirecionar para uma nova página ao invés disso deve-se configurar o servidor para que execute o redirecionamento de forma transparente ao usuário. Utilize o código apropriado HTTP(301). O uso de cabeçalhos HTTP é preferível, porque reduz o tráfego de Internet e os tempos de *download*, pode ser aplicado a documentos “non-HTML”, e pode ser utilizado por agentes que apenas fazem uso do HEAD (e.g., verificadores de *links*). Os códigos de status do tipo 30x, também fornecem informação como “*moved permanently*” (movido permanentemente) ou “*moved temporarily*” (movido temporariamente) os quais não podem ser dados pelo *refresh* do META.

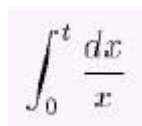
Como solução menos indicada pode-se substituir a página que vai ser redirecionada, por outra estática que contenha um *link* normal para a nova página.

Recomendação 2.6 - Não provocar o aparecimento de janelas de sobreposição, janelas *popup* ou outras quaisquer, assim como nenhuma modificação do conteúdo sem que o usuário seja informado disso. Não é recomendável para o usuário, que *links* abram em uma nova janela. Se o usuário utiliza um navegador com tela cheia não poderá voltar para a página anterior, também o histórico e a possibilidade de ir e voltar a páginas visitadas ficam comprometidos. Caso você tenha uma real necessidade que sua página abra uma outra janela, informe ao usuário:

```
<a href="outroSiteo.htm" target="_blank">Link para outro sítio, que se abrirá em uma nova janela do navegador.</a>
```

Recomendação 2.7 - Sempre que existir uma linguagem de marcação apropriada, utilizar marcações em vez de imagens para transmitir informações, um exemplo é a linguagem MathML que permite a criação de fórmulas matemáticas somente utilizando-se das *tags* apropriadas.

Exemplo: No caso de equações matemáticas use MathML para sua construção e folhas de estilo para formatar o texto e controlar o *layout*. Evite também usar imagens para representar texto.


$$\int_0^t \frac{dx}{x}$$

No código html:

```
<semantics>
<mrow> <msubsup>
<mo>∫</mo>
<mn>0</mn>
<mi>t</mi>
</msubsup>
<mfrac> <mrow>
<mo>d</mo>
<mi>x</mi></mrow>
<mi>x</mi>
</mfrac></mrow>
<annotation-xml encoding="MathML-Content">
<apply><int/>
<bvar><ci>x</ci></bvar>
<lowlimit><cn>0</cn></lowlimit>
<uplimit><ci>t</ci></uplimit>
<apply><divide/>
<cn>1</cn>
<ci>x</ci>
</apply></apply>
</annotation-xml>
</semantics>
```

Existem programas que convertem equações para MathML. Para maiores informações sobre este formato acesse a página do W3 Consortium: <http://www.w3.org/TR/REC-MathML/>.

Recomendação 2.8 - Utilizar o elemento “blockquote” para marcar citações quando existentes. Não use “blockquote”, “ul”, “dl” & “dt”, “table” e outros elementos para criar efeitos visuais nos parágrafos. Caso a intenção seja organizar a estrutura ou a disposição de textos no sítio, utilize folhas de estilo.

A Arte e a Vida

A arte baseia-se na vida, porém não como matéria mas como forma. Sendo a arte um produto directo do pensamento, é do pensamento que se serve como matéria; a forma vai buscá-la à vida. A obra de arte é um pensamento tornado vida: um desejo realizado de si-mesmo. Como realizado tem que usar a forma da vida, que é essencialmente a realização; como realizado em si-mesmo tem que tirar de si a matéria em que realiza.

Fernando Pessoa, in 'Ricardo Reis - Prosa'

No código html:

```
<blockquote>  
<p> <strong>A Arte e a Vida </strong></p>
```

```
<p> A arte baseia-se na vida, porém não  
como matéria mas como forma. Sendo a arte  
um produto directo do pensamento, é do  
pensamento que se serve como matéria; a  
forma vai buscá-la à vida. A obra de arte é um  
pensamento tornado vida: um desejo realizado  
de si-mesmo. Como realizado tem que usar a  
forma da vida, que é essencialmente a  
realização; como realizado em si-mesmo tem  
que tirar de si a matéria em que realiza.  
</p>
```

```
<p><em>Fernando Pessoa, in 'Ricardo Reis -  
Prosa'</em></p>  
</blockquote>
```

Use folha de estilo para controlar o *layout* das páginas, formatar cor, tamanho e tipo de fonte. Os elementos “font” e “basefont” estão ultrapassados no HTML 4.0. Não use os elementos “h1” a “h6” para formatar os textos.

Recomendação 2.9 - Criar um estilo de apresentação coerente e sistemático, ao longo das diferentes páginas, como exemplo, mantendo um padrão de desenho, agrupando os itens do menu de forma coerente. Mantenha os botões principais de navegação no mesmo local em cada página. Isso ajudará ao usuário a localizar-se rapidamente, e saber o destino de cada botão levará. Mantenha para o sítio uma paleta de cores, estilos de texto e diagramação consistente. Identifique as regiões da página, navegação e fim de página de forma clara.

Recomendação 2.10 - Utilize elementos de cabeçalho de forma lógica, organizando o conteúdo de acordo com uma hierarquia.

O Título

Capítulo 1

Aqui vai um texto introdutório.

Uma subdivisão do capítulo

O resto do texto...

No código html:

```
<h1>O Título</h1>
```

```
<h2>Capítulo 1</h2>
```

```
<p> Aqui vai um texto introdutório.</p>
```

```
<h3>Uma subdivisão do capítulo</h3>
```

```
<p>O resto do texto...</p>
```

Recomendação 2.11 – Sempre que necessário, divida grandes blocos de informação em grupos mais fáceis de gerenciar. As opções de menu devem ser dispostas de forma consistente na mesma ordem relativa no grupo de opções. Se as opções num painel de menu estão ordenadas “arquivo, editar, inserir, imprimir”, essas opções devem aparecer naquela mesma ordem quando aquele grupo for apresentado novamente (ou quando um outro painel contendo aquele mesmo grupo de opções seja apresentado).

Em HTML, use “optgroup” para agrupar os elementos “option” dentro de um elemento de lista “select”; agrupe os controles de formulário com “fieldset” e “legend”. Use listas sempre que seja apropriado. Use cabeçalhos para estruturar documentos, etc.

Uma lista do tipo “form select” com sete elementos terá o seguinte aspecto (i.e, uma lista estruturada simples):

Servidor 3 na porta 3.7.1

Servidores e Portas
Servidor 3
Servidor 3 na porta 3.7.1
Servidor 3 na porta 3.7
Servidor 3 na porta 3.5

Servidor 2
Servidor 2 na porta 3.7
Servidor 2 na porta 3.5

Linux
Servidor 1
Servidor 1 na porta 3.7x
Servidor 1 na porta 3.5x

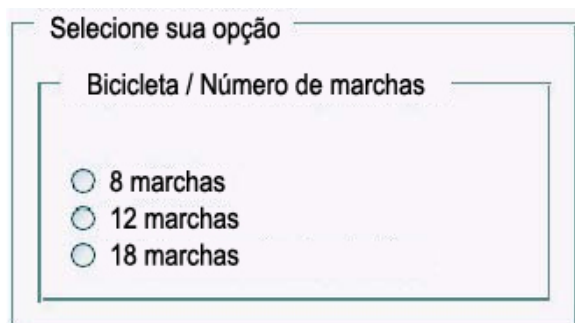
No código html:

```
<select name="Servidores">  
<optgroup label="Portas e Servidores">  
<optgroup label="Servidor 3">  
<option label="3.7.1" value="sp3_3.7.1">  
Servidor 3 na porta 3.7.1  
<option label="3.7" value="sp3_3.7">  
Servidor 3 na porta 3.7  
<option label="3.5" value="sp3_3.5">  
Servidor 3 na porta 3.5  
</optgroup> <optgroup label="Servidor 2">  
<option label="3.7" value="sp2_3.7">  
Servidor 2 na porta 3.7  
<option label="3.5" value="sp2_3.5">  
Servidor 2 na porta 3.5  
</optgroup> </optgroup>  
<optgroup label="Linux">  
<optgroup label="Servidor 1">  
<option label="3.7x" value="sp_3.7x">  
Servidor 1 na porta 3.7x  
<option label="3.5x" value="sp_3.5x">  
Servidor 1 na porta 3.5x  
</optgroup> </optgroup>  
</select>
```

O elemento "fieldset" é utilizado para agrupar controles de formulário relacionados de maneira lógica. Os dispositivos de apoio ou agentes devem fornecer sentido aos formulários complexos através de navegação apropriada dentro do grupo ou grupos. Um navegador gráfico deve mostrar esta relação através de uma borda em volta dos elementos relacionados.

Ao associar o atributo "legend" de forma significativa para cada grupo "fieldset", o autor fornece uma legenda que explica claramente do propósito ou natureza dos agrupamentos. Os agentes

de usuário que suportem o “legend” serão capazes de tornar essa informação disponível para quem deseja. Alguns navegadores gráficos atuais mostram a legenda como texto que surge num espaço circunscrito por um borda pelo elemento “fieldset”.



No código html:

```
<fieldset> <legend> Selecione sua opção
</legend> </fieldset>
<legend>Bicicleta /Número de marchas</legend>
<br>
<input type="radio" name="marchas" value="8m">
8 marchas <br>
<input type="radio" name="marchas" value="12m">
12 marchas<br>
<input type="radio" name="marchas" value="18m">
18 marchas
</fieldset> </fieldset>
```

3.2.2 No caso de serem utilizadas tabelas

Recomendação 2.12 - Não utilizar tabelas para efeitos de disposição em página, prefira o uso de folhas de estilo para a diagramação das páginas. Sendo utilizadas tabelas construa de forma que a disposição continue a fazer sentido depois de ser linearizada. Em último caso, forneça um equivalente alternativo (que pode ser uma versão linearizada).

Recomendação 2.13 - Se for utilizada uma tabela para efeitos de disposição em página, não utilizar qualquer marcação estrutural para efeitos de formatação visual. Não use comandos destinados a indicar cabeçalhos de tabela como o “th” para formatar parágrafos ou fazer títulos em “bold”.

Correto:
<td>Dê-me texto em negrito</td>

Errado:
<th>Dê-me texto em negrito</th>

3.2.3 No caso de serem utilizados formulários

Recomendação 2.14 - Incluir caracteres pré-definidos de preenchimento nas caixas de edição e nas áreas de texto, até que os navegadores tratem corretamente os controles vazios.

Alguns navegadores antigos não permitem que a tecla TAB seja usada para movimentação dentro de formulários. Para isso, coloque um texto no campo do formulário para que o campo seja localizado de forma mais fácil.

Insira seu nome:

Comentário:

No código html:

```
Insira seu nome: <input type="text"
name="name" size="50" value="|"><br><br>
```

```
Comentário: <textarea name="textarea1"
rows="4" cols="50">Introduza por favor os seus
comentários aqui: </textarea><br><br>
```

```
<input type="submit" name="Submit"
value="Submeter este formulário">
```

Recomendação 2.15 - Usar o elemento “label” juntamente com o atributo “id” para associar os rótulos aos respectivos controles dos formulários. Assim, os leitores de tela associarão os elementos do formulário de forma correta. Usando o comando “label” as pessoas que usam leitores de tela não terão problemas ao ler o formulário. Caso haja grupos de informação, controles, etc, a estes devem estar devidamente diferenciados, seja por meio de espaçamento, localização ou elementos gráficos.

Em uma aplicação, os rótulos dos campos são posicionados, consistentemente, à esquerda do campo mostrado, os rótulos dos ícones posicionados abaixo do ícone mostrado e rótulos para botões de rádio são posicionados, consistentemente, à direita.

Nota: Atenção especial deve ser dispensada na diferenciação visual entre rótulo e informação colocados próximos uns dos outros.

Opções de Turno
<input type="checkbox"/> Turno da Manhã
<input type="checkbox"/> Turno da Tarde
<input type="checkbox"/> Turno da Noite
<input type="checkbox"/> Turno da Madrugada

No código html:

```
<fieldset>
<legend align="top">Opções de Turno</legend>
<input type="checkbox" name="opt1" id="opt1"
value="mn">
<label for="opt1"> Turno da Manhã</label><br>
<input type="checkbox" name="opt2" id="opt2" value="td">
<label for="opt2">Turno da Tarde </label><br>
<input type="checkbox" name="opt3" id="opt3" value="nt">
<label for="opt3"> Turno da Noite</label><br>
<input type="checkbox" name="opt4" id="opt4"
value="md">
<label for="opt4">Turno da Madrugada</label><br>
</fieldset>
```

Caso o formulário seja em duas colunas, os rótulos de tamanhos significativamente diferentes devem estar alinhados à direita com os campos alinhados à esquerda, em todos os formulários do sistema.

Recomendação 2.16 - Assegurar o correto posicionamento de todos os controles de formulários que tenham rótulos implicitamente associados, até que os leitores de tela ou navegadores do suporte associações explícitas entre rótulos e controles de formulários.

Associe legendas aos controles dos formulários de forma que a informação seja clara, a legenda deve estar está imediatamente ao lado do controle e posicionada na mesma linha.

[verde | amarelo | azul] **No código html:**

```
[<input type=radio id=verde><label for=verde>verde</label>
<input type=radio id=amarelo><label for=amarelo>amarelo
</label>
<input type=radio id=azul><label for=azul> azul </label>]
```

Recomendação 2.17 – Forneça informações sobre como o sítio está estruturado, através de um mapa ou de sumário. Crie o mapa de forma textual, associando aos títulos das páginas para não causar confusão aos usuários que utilizarem leitores de tela.

O mapa ou sumário do sítio pode mostrar a hierarquia das páginas, isso é feito utilizando o atributo “title” dentro do comando “a”.

[Página de boas-vindas](#) [Página de entrada](#) (página principal) [Produtos](#) [Sistemas](#) [Sistema 1000](#)

No código html:

```
<a href="a.htm" title="nível 1">Página de boas-vindas</a>  
<a href="b.htm" title="nível 2">Página de entrada</a> (página principal)  
<a href="c1.htm" title="nível 3">Produtos</a>  
<a href="d1.htm" title="nível 4">Sistemas</a>  
<a href="e1.htm" title="nível 5">Sistema 1000</a>
```

3.2.4 No caso de serem utilizados applets e programas interpretáveis

Recomendação 2.18 - Assegure a acessibilidade de objetos programados, tais como programas interpretáveis e *applets*, garantindo que a resposta a eventos seja independente do dispositivo de entrada e que qualquer elemento dotado de interface própria possa funcionar com qualquer leitor de tela ou navegador que o usuário utilize. Evite colocar scripts que estejam vinculados a *links*. Se isso não for possível, forneça informações equivalentes em uma página alternativa acessível.

Se um *applet* (geralmente criado por “object”, “flash” ou “applet”) requerer a interação do usuário que não possa ser duplicada num formato alternativo, recomenda-se que faça o *applet* acessível diretamente.

A acessibilidade de objetos com a sua própria interface é independente da acessibilidade do agente de usuário. A acessibilidade deve por isso ser construída nos objetos ou ser fornecida de forma alternativa. Se você é programador, deve estar consciente dos recursos disponíveis que o ajudam a verificar se os seus programas são acessíveis.

Mantenha-se informado sobre as últimas atualizações das linguagens utilizadas provavelmente elas evoluirão de forma a contemplar a acessibilidade.

Exemplo com um objeto em Flash:

```
<object alt="Animação com o logo" title="Animação com o logo">  
<param name=movie value="abertura.swf"> ...  
<script language=javascript>  
<!--...!--></script> <noscript>  
 </noscript> <noembed>  
 </noembed>  
Animação com o logo</object>  
<div id="Abertura" style="visibility: hidden;" title="Descrição da animação"></div>
```

Nem todas os usuários têm computadores que disponham de *mouse* ou outros dispositivos parecidos. Alguns usuários dependem do teclado comum, teclados alternativos ou microfone para utilizar o computador. Por exemplo, o acesso via teclado a *links* e controles de formulário pode ser especificado de algumas maneiras, tais como:

Por atalhos de teclado: Atalhos de teclado permitem aos usuários combinar teclas de atalho para navegar nos *links* e nos controles de formulário numa página. As teclas de atalho podem ser modificadas de acordo com os diferentes sistemas operacionais, residindo a diferença essencialmente na seqüência de teclas a executar para ativar o atalho. Nas máquinas com sistema Windows, as teclas "alt" e "ctrl" são vulgarmente as teclas eleitas; no Macintosh, as teclas "maçã" ou "command".

Pela ordem dos Tabs: A ordem dos "tabs" descreve a (logicamente) ordem de navegação de *link* para *link* ou de um campo de formulário para outro campo, usualmente pressionando a tecla "tab". Teste o seu sítio navegando somente com a tecla "tab", para avançar de *link* em *link*, para retornar utilize "shift + tab".

Embutindo-se nas interfaces de *software*, controles independentes de dispositivos de *hardware*: Alguns elementos existentes em objetos cujas interfaces não possam ser controladas por nenhuma linguagem de notação. Por exemplo, em HTML: *applets*, leitores de multimídia ou Flash. Os desenvolvedores devem certificar-se que os objetos importados (por si só) forneçam interfaces acessíveis; ou que, pelo menos, exista uma alternativa que o faça.

Recomendação 2.19 - Em programas interpretáveis, especificar respostas a eventos, preferindo as rotinas dependentes de dispositivos (*mouse*, teclado, etc).

- Use “onmousedown” com “onkeydown”.
- Use “onmouseup” com “onkeyup”
- Use “onclick” com “onkeypress”
- Use “onfocus” com “onmouseover”
- Use “onblur” com “onmouseout”

Note que não existe equivalente de teclado para duplo-click (“ondblclick”) em HTML 4.0.

Utilize eventos independentes do dispositivo, que trabalhem com qualquer ação de entrada apropriada, no caso, “onfocus” trabalha com o teclado e com o *mouse*, e é por isso considerado um evento independente do dispositivo.

Exemplo:

Código do botão: `<button onfocus="pop1()"> Ganhe um prêmio!</button>`

O script:

```
<script language="javascript" type="text/javascript"> function pop1() { alert  
("Primeiro Prêmio!") }</script>  
<noscript>Primeiro Prêmio!</noscript>
```

O exemplo acima o botão é ativado com o *script*, ao clicar nele ou usar TAB uma pequena caixa de alerta surgirá contendo uma mensagem sobre o prêmio. Pressionando a tecla “escape” fará com que a mesma desapareça.

Em alguns casos deve-se usar mais de um evento, quando este for dependente de dispositivo: “onclick” requer mouse, enquanto que “onkeypress” requer teclado.

Ambos são dependentes de dispositivos, mas a sua inclusão vai permitir a qualquer um beneficiar-se da técnica.

Código do botão: `<button onclick="pop2()" onkeypress="pop2()">
Ganhar um prêmio diferente!</button>`

Código da função *script*:

```
<script language="javascript" type="text/javascript">function pop2() { window.open  
(premio.htm) } </script>
```

```
<noscript><a href="cadastro.htm">Cadastre-se agora!</a></noscript>
```

Neste exemplo o botão é tanto ativado pelo comando de teclado quanto pelo clique do mouse.

3.3 Nível de Prioridade 3

Normas e Recomendações de Acessibilidade que sendo implementadas facilitarão o acesso aos documentos armazenados na Web. Se não cumpridas, grupos de usuários poderão encontrar dificuldades para acessar as informações dos documentos armazenados na Web.

3.3.1 Casos Gerais

Recomendação 3.1 - Não usar elementos considerados ultrapassados pelo W3C.

Evite usar comandos que caíram em desuso pelo HTML 4.0 como “blink”, “marquee”, “applet”, “basefont”, “center”, “dir”, “align”, “font”, “isindex”, “menu”, “strike”, “u”. Para as formatações dê preferência ao uso de folhas de estilo. Os comandos “listing”, “plaintext” e “xmp” que podem ser substituídos pelo comando “pre”.

Recomendação 3.2 - Especificar por extenso cada abreviatura ou sigla, quando da sua primeira ocorrência em um documento, utilizando os atributos “abbr” e “acronym”. Utilize o atributo “abbr” dentro de um elemento “th” quando você tiver cabeçalhos muito longos, para que os leitores de tela lerem apenas o seu conteúdo e não o texto do cabeçalho na íntegra.

No código html:

Quando estiver em Boston, não se esqueça de visitar o MFA, o MIT e, claro, o W3C. Pode chegar facilmente a estes destinos através da Av. Mass ou Mem. Dr.

Quando estiver em Boston, não se esqueça de visitar o <acronym title="Museum of Fine Arts">MFA</acronym>, o <acronym title="Massachusetts Institute of Technology">MIT</acronym> e, claro, o <acronym title="World Wide Web Consortium">W3C</acronym>. Pode chegar facilmente a estes destinos através da <abbr title="Avenida Massachusetts">Av. Mass.</abbr> ou <abbr title="Memorial Drive">Mem. Dr.</abbr>

Recomendação 3.3 - Fornecer atalhos por teclado que apontem para *links* importantes (incluindo os contidos em mapas de imagem armazenados no cliente), para início da área principal de conteúdo da página, controles de formulários, e grupo de controles de formulários.

Pode-se permitir que o usuário possa saltar ou ir diretamente a campos do formulário, ou que ele pule o cabeçalho da página indo direto para a área principal de conteúdo da página, utilizando-se do comando “accesskey”.

vai para o fim: Alt + m



No código html:

```
<a href="#fim"> vai para o fim: Alt + m</a>
```

Texto qualquer...

```
<p>texto qualquer</p>
```

FIM



```
<a name="fim" href="#" accesskey="m"> FIM </a>
```

Recomendação 3.4 - Inserir, entre *links* adjacentes, caracteres que não funcionem como *link* e sejam passíveis de impressão (como um espaço), até que os leitores de tela ou navegadores (incluindo as tecnologias de apoio) reproduzam clara e distintamente os *links* adjacentes.

Quando há muitos *links* numa mesma linha, separe-os com caracteres de forma a criar mais espaço entre eles. Isso criará um intervalo entre os *links* quando estiver utilizando um leitor de tela.

[[Seção A](#) | [Seção B](#) | [Seção C](#) | [Seção D](#) | [Seção E](#)]



No código html:

```
[ <a href="a.htm">Seção A</a> | <a href="b.htm">Seção B</a> | <a href="c.htm">Seção C</a> |  
<a href="d.htm">Seção D</a> | <a href="e.htm">Seção E</a> ]
```

Recomendação 3.5 - Sempre que possível, fornecer informações que possibilitem aos usuários receber os documentos de acordo com as suas preferências (por ex., por idioma ou por tipo de conteúdo).

Exemplo: Em vez de incluir *links* tais como “Aqui está a versão francesa deste documento”, use negociação de conteúdos de forma a que a versão Francesa seja apresentada de acordo com o requerido pelos clientes desta versão dos documentos.

- Em HTML 4.0 muitos elementos permitem especificar o idioma com o atributo “lang”.
- Se não for possível usar a negociação de conteúdos, use o “hreflang” em HTML com os elementos “a” ou “link” para identificar o idioma dos documentos alvo.

```
<a href="versao.htm" hreflang="FR"> versão em francês </a>
```

Recomendação 3.6 - Fornecer barras de navegação para auxiliar os menus de navegação. Utilizar elemento que contextualizem a localização do usuário, como barras de caminho e “Sua Localização” nas páginas do documento.

```
Home > Serviços > Consultoria > Investimentos
```

Possuindo um *layout* consistente e uma vez familiarizado, torna-se extremamente fácil navegar pelo conjunto dos *links*. Outro benefício é ser possível saltar de forma rápida para outros conteúdos do documento.

Recomendação 3.7 - Agrupar *links* relacionados entre si, identificando o grupo (em benefício do navegador ou leitor de tela do usuário) e, até que o navegador ou leitor de tela do usuário se encarregue de tal função, fornecer um modo de contornar determinado grupo.

Os usuários cegos saltam freqüentemente de *link* em *link* quando percorrem uma página ou visualizam informação. Quando fazem isto, o “link text” (texto do *link* - texto que se encontra entre as instruções <a> e) é lido pelo leitor de tela. A barra de navegação é, não raras vezes, a primeira coisa a ser encontrada numa página.

Para usuários de sintetizadores de fala, isto significa ter que ouvir um número grande de *links* similares em todas as páginas antes de chegar ao conteúdo singular dessa página. No entanto, quando os *links* se encontram agrupados em conjuntos lógicos, tais como barras de navegação, os sintetizadores conseguem manipular uma peça em vez de diversas peças. Desta forma, eles podem preceder os elementos agrupados por um *link*, permitindo ao usuário saltar o conjunto de *links* e posicionar-se de forma imediata no início do corpo principal da página. Uma forma de agrupar os *links* é usando o comando MAP. Não é necessário ter uma imagem associada ao comando para poder usá-lo.

Navegar no sítio.

[[Seção A](#) | [Seção B](#) | [Seção C](#) | [Seção D](#)]



No código html:

```
<map name="map2">  
Navegar no sítio.<br>  
[ <a href="a.htm">Seção A</a> |  
<a href="b.htm">Seção B</a> |  
<a href="c.htm">Seção C</a> |  
<a href="d.htm">Seção D</a> ]  
</map>
```

Recomendação 3.8 - Se forem oferecidas funções de pesquisa, ativar diferentes tipos de pesquisa de modo a corresponderem a diferentes níveis de competência e às preferências dos usuários. Sendo possível, quando a pesquisa não encontrar a palavra, sugerir palavras semelhantes.

Exemplo: Essa recomendação não obriga você a colocar um mecanismo de pesquisa no sítio. Mas, ao colocar, faça de maneira acessível. É importante conhecer os tipos de pesquisa (busca) que existem, para quando implementar no sítio, escolher o mais adequado ao seu projeto.

- Pesquisa por palavra-chave: Neste tipo de pesquisa o usuário digita apenas uma palavra que o mecanismo de pesquisa procura em todo o sítio, por páginas que contenham a palavra digitada.

- Pesquisa complexa: Nesta pesquisa, além de digitar uma palavra, o usuário adiciona operadores Booleanos (*and*, *or*). Caso a busca não encontre, sugira palavras semelhantes.

Recomendação 3.9 - Use palavras relevantes no início de cabeçalhos, parágrafos, e listas para identificar o assunto tratado.

Inicie cabeçalhos, parágrafos, listas, com uma informação distinta. Isto é vulgarmente referenciado por “front-loading” (à frente) e é especialmente útil para o acesso à informação de forma serial, reduzindo o esforço é necessário para compreensão e localização de informações importantes para o usuário.

Uma lista que usa o método ‘front-load :

- [Starfish e Lobsters, e Crabs, Oh My!](#) é sem dúvida o melhor sítio.
- [Sea Cucumbers Galore](#) é outro dos sítios sobre invertebrados com interesse para uma visita.
- [Visite o maravilhoso mundo da esponja](#), se tiver um momento.

Uma lista que não usa o método “front-loaded”:

- Mas o melhor sítio em termos absolutos é o [Starfish e Lobsters, e Crabs, Oh My!](#)
- Outro sítio sobre invertebrados é o [Sea Cucumbers Galore](#).
- Se tiver um momento, [visite o Maravilhoso Mundo da Esponja](#), se estiver voltado para o tema.

Recomendação 3.10 - Fornecer informações sobre documentos compostos por várias páginas (isto é, coleções de documentos). Caso seja necessário, utilize ferramentas de compactação de arquivo, tais como ZIP, TAR, GZIP ou ARJ. Informe o tamanho do arquivo e o tempo estimado para baixar por meio de um modem comum. Forneça documentos em formatos alternativos, passíveis de leitura pelos leitores de tela.

Deve-se indicar o número de páginas existentes num documento seqüência html, como manuais ou instruções passo-a-passo e sua navegação deve ser colocada de forma clara.

```
<head>  
<link rel="anterior" href="chk9-0.htm">  
<link rel="índice" href="overchk.htm">  
<link rel="próximo" href="chk11-0.htm">  
</head>
```

Acima temos um exemplo que mostra uma forma de fornecer informação sobre um documento seqüencial. Ele mostra como se usa a notação “link rel=next” e “link rel=prev” para indicar o URL da próxima página e da anterior como informação de navegação extra.

```
<p>O link a seguir se refere ao manual das recomendações (pontos  
de verificação) do W3C. Possui 150Kb e seu tempo estimado para  
download é de 60 segundos. <br>  
<a href="manual.zip"> Manual do W3C (150kb formato .rtf  
compactado </a></p>
```

Os documentos disponibilizados para *download* devem estar em formatos compatíveis com os leitores de tela, senão, disponibilizar também na página o *download* de uma versão compatível textual (ex: formatoTXT).

Recomendação 3.11 - Complementar o texto com apresentações gráficas ou sonoras, sempre que puderem facilitar a compreensão da página.

Em certos casos, torna-se necessária a utilização de equivalentes não textuais, tais como imagens, animações ou vídeos. Isto é especialmente útil para os analfabetos que podem visualizar as apresentações visuais, para os surdos que possuem comunicação com as mãos (língua gestual) e para os analfabetos que também podem se beneficiar muito de equivalentes gráficos.

Os equivalentes não visuais e não textuais são bastante diversos. Entre os mais comuns, encontra-se a pré-gravação, áudio de música, língua falada, ou efeitos sonoros. Esses equivalentes serão especialmente importantes para os analfabetos que podem perceber as

apresentações em áudio. As apresentações pelo suporte de áudio produzidas por sintetizadores de fala, e com suporte tátil de Braille, são geralmente derivadas do texto, ou descrições.

Recomendação 3.12 - Identificar claramente o destino de cada *link*, botão ou elemento que submeta uma ação. Prefira utilizar textos mais claros e objetivos, mostrando o verdadeiro sentido e o destino do *link*. Evite usar frases como “Clique aqui”.

O texto do *link* deve ser facilmente compreensível e conciso para que tenha sentido quando for lido, mesmo fora do dispositivo padrão. Como por exemplo, um leitor de tela.

Exemplo correto:

[Ganhe um prêmio](#) fornecido pelo nosso patrocinador.

Exemplos incorretos:

[Ganhe um prêmio fornecido pelo nosso patrocinador.](#)

[Clique aqui](#) para ganhar um prêmio de nosso patrocinador.

No exemplo correto a frase está colocada de forma concisa. Desta maneira, o *link* será compreensível até para usuários de leitores de tela. Nos exemplos incorretos, fazer o link em toda a extensão da frase não é necessário e pode causar confusão para quem usa os leitores de tela. Já no segundo exemplo, a frase “Clique aqui” fica muito vaga. Onde é o “aqui” da frase? Outro recurso é a inclusão do atributo “title” dentro do *link* para dar mais informação ao usuário.

```
<a href="next.htm" title="Tudo sobre receita fiscal.">Receita fiscal</a>
```

Recomendação 3.13 - Informar previamente ao usuário o destino e resultado da ação, quando houver campos e elementos do formulário, como, por exemplo, caixas de seleção, que submetem automaticamente o conteúdo ao se efetuar uma determinada seleção. Nestes casos, ao invés da seleção submeter automaticamente o formulário, é recomendável que se vincule ao elemento um botão para efetuar a ação;

Não havendo a possibilidade de colocar-se um botão para ação posterior - antes de submeter o formulário, notificar o usuário que uma ação deste tipo acontecerá, assim como o tempo

estimado para o processamento e a construção da nova página.

Ainda, quando da carga da nova página, utilizar um menu “escondido/invisível” no início da mesma, possibilitando que o usuário receba uma mensagem de que a página foi recarregada e que possa escolher uma opção que o leve diretamente para o conteúdo onde estava anteriormente (como um atalho, por exemplo).

Recomendação 3.14 - Forneça metadados para acrescentar informações semânticas e descritivas do sítio, que sejam úteis para os mecanismos de busca.

Pode-se usar o comando “link rel” para criar relações entre os documentos. Esse tipo de associação é interessante para usuários de LINX (navegador texto).

```
<head>  
<link rel="anterior" href="chk9-0.htm">  
<link rel="índice" href="overchk.htm">  
<link rel="próximo" href="chk11-0.htm">  
</head>
```

4 Relação entre as Áreas de Acessibilidade compreendidas na Visão do Cidadão e as Recomendações da Visão Técnica

No documento Modelo de Acessibilidade é descrito a visão do cidadão, uma forma de entendimento do modelo com foco no cidadão, e não no desenvolvedor. A Visão do Cidadão tem um público mais abrangente que inclui pessoas não técnicas, com uma perspectiva de compreensão mais intuitiva quanto aos resultados do processo de acessibilidade.

As Recomendações de Acessibilidade podem ser segmentadas quanto à Visão do Cidadão auxiliando na compreensão de onde cada recomendação contribui no resultado percebido pelo cidadão sendo divididas a partir de suas quatro áreas: Área da Percepção, Área da Operação, Área do Entendimento e Área da Compatibilidade.

Da mesma forma que as Diretrizes da Visão Técnica organiza as recomendações de forma a facilitar o domínio do técnico agrupando as recomendações de acordo com a percepção do resultado, as recomendações dividem-se de acordo com as Áreas da Visão do Cidadão, consolidando assim a relação existente entre as duas visões, auxiliando na compreensão de onde cada recomendação contribui no resultado percebido pelo cidadão.

Assim relacionamos as recomendações de acordo com as Áreas da Visão do Cidadão:

Área da Percepção: Trata de benefícios relacionados à apresentação do conteúdo, da informação. Ela preocupa-se com a percepção de elementos como gráficos, sons, imagens, multimídia e equivalentes.

Recomendações: 1.11, 1.4, 1.5, 1.6, 1.7, 1.11, 1.12, 1.13, 1.14, 1.15, 1.16, 1.17, 1.18, 1.20, 1.21, 1.22, 1.23, 2.2, 2.3, 2.8, 2.9, 2.10, 2.11, 2.12, 2.13, 2.17, 3.1, 3.4, 3.11.

Área da Operação: Preocupa-se com a manipulação da informação, do conteúdo. Ou seja, a Área da Operação deve garantir formas alternativas ao acesso às informações através de maneiras diferenciadas de navegação ou técnica similar. Percebe-se, também, que é de responsabilidade da Operação garantir sempre ao usuário o controle da navegação e interação com o sítio.

Recomendações: 1.3, 1.4, 1.8, 1.10, 1.15, 1.16, 1.17, 1.20, 1.23, 1.24, 2.1, 2.4, 2.5, 2.6, 2.11, 2.14, 2.15, 2.16, 2.19, 3.1, 3.3, 3.4, 3.6, 3.7, 3.8, 3.10, 3.13.

Área do Entendimento: Essa, por sua vez, trata de questões relacionadas à compreensão do conteúdo publicado. Ela deve garantir que todo o conteúdo apresentado seja de fácil compreensão para qualquer tipo de usuário.

Recomendações: 1.11, 1.1, 1.2, 1.9, 2.7, 2.11, 2.15, 3.2, 3.5, 3.7, 3.9, 3.10, 3.11, 3.12.

Área da Compatibilidade: Aborda questões como a necessidade de utilizarmos sempre de tecnologias acessíveis e compatíveis com o modelo aqui proposto.

Recomendações: 1.15, 1.16, 1.19, 1.20, 2.18, 3.1.

5 Detalhes da Avaliação e Validação

Baseando-nos no que foi preconizado em *Detalhes da Avaliação e Validação* no documento “eMAG, Acessibilidade de Governo Eletrônico – Modelo”, percebemos que o processo de avaliação e validação da acessibilidade deve passar por três (3) etapas distintas. Para facilitar a realização de tais etapas, abaixo citamos alguns tópicos de relacionados.

É importante deixar claro que as diretrizes por si só não são capazes de garantir a acessibilidade. As recomendações apenas orientam para que os requisitos de acessibilidade sejam cumpridos. É importante que o sítio seja avaliado e testado em avaliadores de acessibilidade e programas específicos para pessoas portadoras de necessidades especiais.

5.1 Programas Avaliadores de Acessibilidade

Existem programas na Web que avaliam o nível de acessibilidade em sítios na Internet. Tais programas produzem relatórios precisos com os problemas encontrados e que deveriam ser corrigidos para que o sítio se torne acessível.

A maior parte dos programas aqui relacionados trabalha com base no W3C/WAI, sendo o “Da Silva”, desenvolvido no Brasil que contempla, além do W3C/WAI, o modelo de níveis conforme proposto por este documento.

Em português:

Da Silva: <http://www.acessobrasil.org.br>;

Em inglês:

Bobby (empresa norte-americana Watchfire Corporation): <http://bobby.watchfire.com>;

Cynthia Says: <http://www.cynthiasays.com>;

Lift (empresa Usablenet): <http://www.usablenet.com>.

Em espanhol:

TAW: <http://www.tawdis.net>;

5.2 Programas Leitores de Tela

Os usuários com necessidades especiais utilizam-se de programas capazes de ler e interpretar diretamente a tela do computador. No Brasil, são utilizados três leitores de tela:

Dosvox/Webvox (desenvolvido no Brasil): <http://intervox.nce.ufrj.br>;

Virtual Vision (desenvolvido no Brasil): <http://www.micropower.com.br>;

Jaws for Windows: <http://www.freedomscientific.com>.

Todos estes leitores de tela foram desenvolvidos para o ambiente Windows, mas já existem softwares análogos para Unix, como o Emacspeak – uma interface de áudio para Linux – e o Gnopernicus.

Aconselha-se a utilização de softwares leitores de tela para a realização de testes finais de acessibilidade. Sempre que possível recomenda-se a utilização de usuários com necessidades especiais para efetuar testes nas páginas do sítio.

6 Leitura complementar

Gerais

Acessibilidade Brasil - www.acessobrasil.org.br

Web Accessibility Initiative - www.w3.org/WAI

Acessibilidade.net - www.acessibilidade.net

Dicas para fazer sites de Web acessíveis

WAI QuickTips - <http://www.w3.org/WAI/References/QuickTips/qt.pt.htm>

Lista de ferramentas para acessibilidade - www.w3.org/WAI/ER/existingtools.html

Techniques For Accessibility Evaluation And Repair Tools - <http://www.w3.org/TR/AERT>

Side by Side WCAG vs. 508 - Web <http://jimthatcher.com/sidebyside.htm>

IBM Accessibility Center - <http://www-306.ibm.com/able/>

SUN Accessibility - <http://www.sun.com/access/>

Microsoft Accessibility - <http://www.microsoft.com/enable/>

Apple Accessibility - <http://www.apple.com/accessibility/>

Introdução à criação de um site acessível - www.diveintoaccessibility.org

Rede SACI - www.saci.org.br

PRODAM - www.prodam.sp.gov.br/acess

WebAIM - www.webaim.org

Checklist de Acessibilidade para Usuários Idosos - www.labiutil.inf.ufsc.br/acessibilidade

Países com leis de acessibilidade

Alemanha - www.bmgs.bund.de/nn_617014/EN/Social-Security/Disabled-persons/disabled-persons-node.param=.html_nnn=true

Austrália - www.hreoc.gov.au/disability_rights/standards/standards.html

Canadá - www.tbs-sct.gc.ca/ig-gi

Dinamarca - www.fsk.dk/fsk/publ/1997/freedom

Espanha - www.sidar.org/index.php

Estados Unidos - www.access-board.gov/sec508/guide/

França - www.senat.fr/accessibilite.html

Hong Kong - www.info.gov.hk/digital21/e-gov/eng/roadmap/a.htm

Índia - www.kspfc.org/right-to-information.htm

Ireland - www.accessit.nda.ie/policy_and_legislation.html

www.accessit.nda.ie

Itália - www.pubbliaccesso.it/normative/law_20040109_n4.htm

www.innovazione.gov.it/ita/intervento/accessibilita.shtml

Japão - www.soumu.go.jp/joho_tsusin/eng

Nova Zelândia - www.e-government.govt.nz/web-guidelines

Portugal - www.acessibilidade.net/petition/government_resolution.html

www.acesso.unic.pcm.gov.pt/

Reino Unido - www.cabinetoffice.gov.uk/e-government/resources/eaccessibility/index.asp

União Européia - europa.eu.int/information_society/policy/accessibility