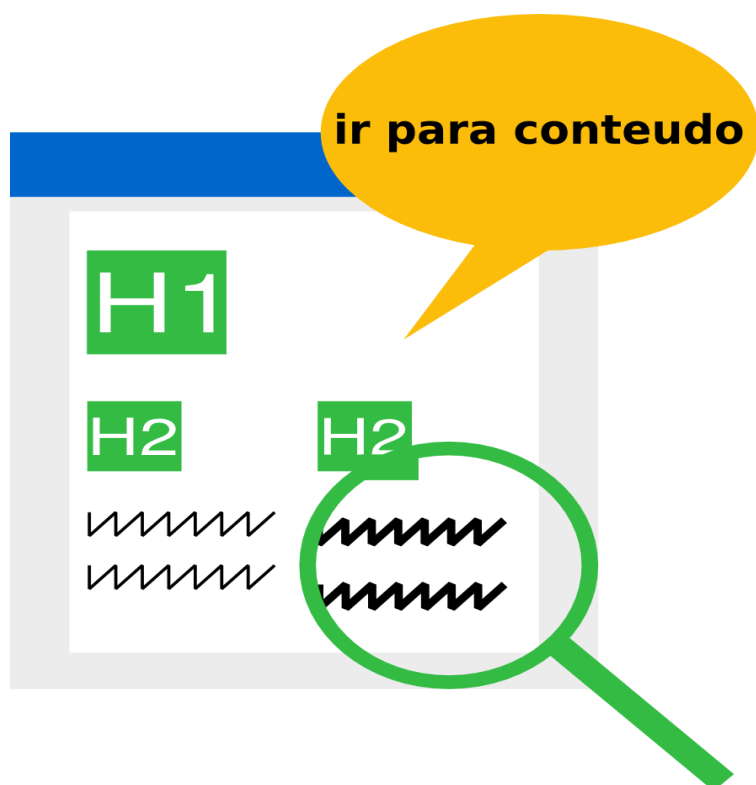


Ministério do Planejamento, Orçamento e Gestão
Secretaria de Logística e Tecnologia da Informação
Departamento de Governo Eletrônico
www.governoeletronico.gov.br

Ministério da Educação
Secretaria de Educação Profissional e Tecnológica
Rede Nacional de Pesquisa e Inovação em Tecnologias Digitais
Projeto de Acessibilidade Virtual



gov.br
e-MAG

Versão 3.0

Modelo de Acessibilidade em Governo Eletrônico

Agosto de 2011



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

e-MAG Modelo de Acessibilidade em Governo Eletrônico/ Ministério do Planejamento, Orçamento e Gestão, Secretaria de Logística e Tecnologia da Informação; Ministério da Educação, Secretaria de Educação Profissional e Tecnológica - Brasília : MP, SLTI, 2011. 69 p.: color.

1. Acessibilidade 2. Sites da Web – Projetos. 3. Sites na Web - Desenvolvimento 4. Programação para Internet 5. Informática - Serviço Público I. Título.

CDU

004.5:35

CDD 004.678



Esta obra está licenciada por uma Licença Creative Commons - Atribuição - Partilha nos Mesmos Termos 3.0 Não Adaptada

<http://creativecommons.org/licenses/by-sa/3.0/br/>

Você tem a liberdade de:

- Compartilhar — copiar, distribuir e transmitir a obra.
- Remixar — criar obras derivadas.

Sob as seguintes condições:

- Atribuição — Você deve creditar a obra da forma especificada pelo autor ou licenciante (mas não de maneira que sugira que estes concedem qualquer aval a você ou ao seu uso da obra).
- Compartilhamento pela mesma licença — Se você alterar, transformar ou criar em cima desta obra, você poderá distribuir a obra resultante apenas sob a mesma licença, ou sob uma licença similar à presente.

Ficando claro que:

- Renúncia — Qualquer das condições acima pode ser renunciada se você obtiver permissão do titular dos direitos autorais.
- Domínio Público — Onde a obra ou qualquer de seus elementos estiver em domínio público sob o direito aplicável, esta condição não é, de maneira alguma, afetada pela licença.
- Outros Direitos — Os seguintes direitos não são, de maneira alguma, afetados pela licença:
- Limitações e exceções aos direitos autorais ou quaisquer usos livres aplicáveis;
- Os direitos morais do autor;
- Direitos que outras pessoas podem ter sobre a obra ou sobre a utilização da obra, tais como direitos de imagem ou privacidade.
- Aviso — Para qualquer reutilização ou distribuição, você deve deixar claro a terceiros os termos da licença a que se encontra submetida esta obra

Agradecimentos

O Departamento de Governo Eletrônico agradece a: Bruna Poletto Salton, Jucélia Poletto Almeida, Agebson Rocha Façanha, André Luiz Rezende, Andréa Poletto Souza, Ângela Guimarães, Felipe Zap, Gleison Samuel do Nascimento, Jorge Fiore de Oliveira Junior, Júlia Marques Carvalho da Silva, Marco Antonio de Queiroz, Marcus Vinicius Bennett Ferreira, Maurício Covolan Rosito, Reinaldo Ferraz, Renato Busatto, Ricardo Moro, Ueslei Paterno e Woquiton Fernandes, que tornaram a versão 3.0 do e-MAG possível.

O DGE também agradece a:

Andre Pimenta Freire
Carol Scarton
Diego Roger Ramos Freitas
Edercio Marques Bento
Jorge Fernandes
Leonelo Dell Anhol Almeida
Maria Cecília Calani Baranauskas
Thiago Prado de Campos
Vagner Figueredo de Santana

Pela sua contribuição, via Consulta Pública, com sugestões, esclarecimentos e correções para o presente documento.

Equipe do DGE – Departamento de Governo Eletrônico

Índice

1. Introdução.....	5
1.1 Sobre a versão 3.0.....	5
1.2 Legislação.....	6
1.3 O acesso de pessoas com deficiência.....	7
1.4 O processo para desenvolver um sítio acessível.....	8
2. Recomendações de acessibilidade.....	12
2.1 Marcação.....	13
2.2 Comportamento (DOM).....	21
2.3 Conteúdo / Informação.....	24
2.4 Apresentação / Design.....	38
2.5 Multimídia.....	44
2.6 Formulários.....	47
3. Padrões de acessibilidade digital no Governo Federal.....	56
3.1 Página de descrição com os recursos de acessibilidade.....	56
3.2 Atalhos de teclado.....	57
3.3 Barra de acessibilidade.....	58
3.4 Apresentação do mapa do sítio.....	59
3.5 Apresentação de formulário.....	60
3.6 Conteúdo alternativo para imagens.....	63
3.7 Apresentação de documentos.....	63
3.8 Elementos que não devem ser utilizados.....	63
4. Recursos e ferramentas para acessibilidade.....	64
5. Leitura complementar.....	66
6. Glossário.....	67
7. Anexo 1 – Tabela de Contraste de Cores	69

1. Introdução

O governo brasileiro, comprometido com a inclusão, buscou, através da elaboração do modelo de acessibilidade do governo eletrônico, facilitar o acesso para todas as pessoas às informações e serviços disponibilizados nos sítios e portais do governo. Assim, a primeira versão do e-MAG, elaborada pelo Departamento de Governo Eletrônico em parceria com a ONG Acessibilidade Brasil, foi disponibilizada para consulta pública em 18 de janeiro de 2005, e a versão 2.0 já com as alterações propostas, em 14 de dezembro do mesmo ano. Em 2007, a Portaria nº 3, de 7 de maio, institucionalizou o e-MAG no âmbito do sistema de Administração dos Recursos de Informação e Informática – SISP, tornando sua observância obrigatória nos sítios e portais do governo brasileiro.

1.1 Sobre a versão 3.0

Para a elaboração da versão 2.0 do e-MAG foi realizado um estudo das regras de acessibilidade através de um método comparativo entre as normas adotadas por diversos países, como a Section 508 do governo dos Estados Unidos, os padrões CLF do Canadá, as diretrizes irlandesas de acessibilidade e documentos de outros países, entre eles Portugal e Espanha. Também foi realizada uma análise detalhada das regras e pontos de verificação do órgão internacional WAI/W3C, presentes na WCAG 1.0.

A versão 2.0 do e-MAG dividia-se em duas partes:

1. a cartilha técnica, voltada a desenvolvedores de sítios, apresentando detalhadamente a proposta de implementação das recomendações de acessibilidade em sítios do governo;
2. a visão do cidadão, voltada a todos os cidadãos brasileiros, apresentando o modelo de acessibilidade de forma simples e de fácil compreensão.

A divisão do e-MAG apresentou alguns inconvenientes durante o processo de disseminação do Modelo, como a dificuldade das pessoas entenderem as áreas da Visão do Cidadão e seu relacionamento com a aplicação efetiva da acessibilidade. O aprendizado durante os seis anos da versão 2.0 do e-MAG e o lançamento da versão 2.0 do WCAG marcaram o caminho para a revisão do Modelo.

A revisão do modelo e a nova versão foi desenvolvida através da parceria entre o Departamento de Governo Eletrônico e o Projeto de Acessibilidade Virtual da RENAPI (Rede de Pesquisa e Inovação em Tecnologias Digitais). Também, para a elaboração dessa nova versão, foram consideradas as contribuições de especialistas na área da acessibilidade.

A elaboração da versão 3.0 foi embasada na versão anterior do e-MAG, apoiando-se na WCAG 2.0, lançada em dezembro de 2008, e considerando as novas pesquisas na área de acessibilidade à Web. Apesar de utilizar a WCAG como referência, o e-MAG 3.0 foi desenvolvido e pensado para as necessidades locais, visando atender as prioridades brasileiras e mantendo-se alinhado ao que existe de mais atual neste segmento.

Seguindo a diretriz do programa de Governo Eletrônico de promover a Cidadania, o documento-proposta passou por Consulta Pública no período de novembro de 2010 a janeiro de 2011, recebendo contribuições tanto pelo sistema de Consulta Pública do Portal do Programa, quanto por e-mail. O número de contribuições superou as expectativas e a avaliação criteriosa destas impactou na data de entrega do modelo, que teve seu cronograma estendido.

Assim, após um longo período de maturação e estudo, é entregue à sociedade brasileira a terceira versão do Modelo de Acessibilidade em Governo Eletrônico, o e-MAG **versão 3.0**, atualizado e mais abrangente no que diz respeito a tornar acessível o conteúdo do governo brasileiro na Web.

A versão 3.0 do e-MAG é apresentada em apenas um documento, não havendo separação entre visão técnica e visão do cidadão. Outra decisão foi o abandono dos níveis de prioridade A, AA e AAA, visto que o padrão é voltado às páginas do Governo, não sendo permitido exceções com relação ao cumprimento das recomendações. Além disso, no e-MAG 3.0 foi incluída a seção chamada “Padronização de acessibilidade nas páginas do governo federal”, com o intuito de padronizar elementos de acessibilidade que devem existir em todos os sítios e portais do governo.

1.2 Legislação

Estão listados abaixo alguns dos principais documentos, que fazem parte da legislação que norteia o processo de promoção da acessibilidade e a implementação do e-MAG:

1. Decreto número 5296, de 2 de dezembro de 2004, que regulamenta as leis nº 10.048, de 8 de novembro de 2000, que dá prioridade de atendimento às pessoas que especifica, e 10.098, de 19 de dezembro de 2000, que estabelece normas gerais e critérios básicos para a promoção da acessibilidade das pessoas com deficiência, e dá outras providências;
2. Comitê CB-40 da ABNT, que se dedica à normatização no campo de acessibilidade, atendendo aos preceitos de desenho universal. O Comitê possui diversas comissões, definindo normas de acessibilidade em todos os níveis, desde o espaço físico até o virtual;
3. Decreto nº 6949, de 25 de agosto de 2009, que promulga a Convenção Internacional sobre os Direitos das Pessoas com Deficiência elaborada pelas Nações Unidas em 30 de março de 2007, definindo, em seu artigo 9º, a obrigatoriedade de promoção do acesso de pessoas com deficiência a novos sistemas e tecnologias da informação e comunicação, inclusive à Internet.
4. Portaria nº 3, de 7 de maio de 2007, que institucionalizou o e-MAG no âmbito do sistema de Administração dos Recursos de Informação e Informática – SISP, tornando sua observância obrigatória nos sítios e portais do governo brasileiro.

1.3 O acesso de pessoas com deficiência

O computador e a Internet representam um enorme passo para a inclusão de pessoas com deficiência, promovendo autonomia e independência. Mas como pessoas com deficiência utilizam o computador? Muitas vezes, a deficiência não é severa o suficiente a ponto de tornar-se uma barreira à utilização do computador. Entretanto, na maioria das páginas da Web, as pessoas cegas ou com baixa visão, pessoas com deficiência auditiva, com dificuldade em utilizar o mouse, por exemplo, encontram inúmeras barreiras de acessibilidade que dificultam ou impossibilitam o acesso aos seus conteúdos.

No que se refere a acesso ao computador, as quatro principais situações vivenciadas por usuários com deficiência são:

- Acesso ao computador sem mouse: no caso de pessoas com deficiência visual, dificuldade de controle dos movimentos, paralisia ou amputação de um membro superior;

- Acesso ao computador sem teclado: no caso de pessoas com amputações, grandes limitações de movimentos ou falta de força nos membros superiores;
- Acesso ao computador sem monitor: no caso de pessoas com cegueira;
- Acesso ao computador sem áudio: no caso de pessoas com deficiência auditiva.

No entanto, esses não são os únicos casos que devem ser considerados quando se pensa em acessibilidade na Web. Muitas pessoas apresentam outras limitações relacionadas à memória, resolução de problemas, atenção, compreensão verbal, leitura e linguística, compreensão matemática e compreensão visual. Uma pessoa com dislexia, por exemplo, pode apresentar dificuldade de leitura de uma página devido a um desenho inadequado. Por isso, um sítio desenvolvido considerando a acessibilidade deve englobar diferentes níveis de escolaridade, faixa etária e pouca experiência na utilização do computador, bem como ser compatível com as diversas tecnologias utilizadas para acessar uma página da Web.

Um dos aliados das pessoas com deficiência para o uso do computador são os recursos de tecnologia assistiva, que auxiliam na realização de tarefas antes muito difíceis ou impossíveis de realizar, promovendo, desta maneira, a autonomia, independência, qualidade de vida e inclusão social de pessoas com deficiência.

Existe atualmente uma enorme gama de recursos de tecnologia assistiva, desde artefatos simples até objetos ou softwares mais sofisticados e específicos, de acordo com a necessidade de cada pessoa. Uma pessoa com limitado movimento das mãos, por exemplo, pode utilizar um teclado adaptado que contém teclas maiores ou um mouse especial para operar o computador. Já as pessoas com baixa visão podem recorrer a recursos como ampliadores de tela, enquanto usuários cegos podem utilizar softwares leitores de tela para fazer uso do computador.

Apesar de sua enorme importância na promoção da acessibilidade às pessoas com deficiência, os recursos de tecnologia assistiva, por si só, não garantem o acesso ao conteúdo de uma página da Web. Para tal, é necessário que a página tenha sido desenvolvida de acordo com os padrões Web (Web Standards) e as recomendações de acessibilidade, os quais serão abordados ao longo deste documento.

Dentro desse contexto, este documento objetiva garantir que o processo de acessibilidade dos sítios do governo brasileiro seja conduzido de forma padronizada, de fácil implementação, coerente com as necessidades brasileiras e em conformidade com os padrões internacionais.

1.4 O processo para desenvolver um sítio acessível

A acessibilidade à Web refere-se a garantir acesso facilitado a qualquer pessoa, independente das condições físicas, dos meios técnicos ou dispositivos utilizados. No entanto, ela depende de vários fatores, tanto de desenvolvimento quanto de interação com o conteúdo. O processo para desenvolver um sítio acessível é realizado em três passos:

1. Seguir os padrões Web;
2. Seguir as diretrizes ou recomendações de acessibilidade;
3. Realizar a avaliação de acessibilidade.

PRIMEIRO PASSO: PADRÕES WEB

Para se criar um ambiente online efetivamente acessível é necessário, primeiramente, que o código esteja dentro dos padrões Web internacionais definidos pelo W3C.

Os padrões de desenvolvimento Web do W3C, ou Web Standards, são um conjunto de recomendações que visa padronizar o conteúdo Web, possibilitando melhores práticas no desenvolvimento de páginas da Web. Uma página desenvolvida de acordo com os padrões Web deve estar em conformidade com as normas HTML, XML, XHTML e CSS, seguindo as regras de formatação sintática. Além disso, é muito importante que o código seja semanticamente correto, ou seja, que cada elemento seja utilizado de acordo com um significado apropriado, valor e propósito.

A conformidade com os padrões Web permite que qualquer sistema de acesso à informação interprete a mesma adequadamente e da mesma forma, seja por meio de navegadores, leitores de tela, dispositivos móveis (celulares, tablets, etc.) ou agentes de software (mecanismos de busca ou ferramentas de captura de conteúdo). Páginas que não possuem um código de acordo com os padrões do W3C apresentam comportamento imprevisível, e na maioria das vezes impedem ou pelo menos dificultam o acesso.

Para conhecer as boas práticas em desenvolvimento de sítios de acordo com os padrões, ver Cartilha de Codificação dos Padrões Brasil e-GOV, disponível em:

<http://www.governoeletronico.gov.br/acoes-e-projetos/padroes-brasil-e-gov>

SEGUNDO PASSO: DIRETRIZES OU RECOMENDAÇÕES DE ACESSIBILIDADE

As diretrizes ou recomendações de acessibilidade explicam como tornar o conteúdo Web acessível a todas as pessoas, destinando-se aos criadores de conteúdo Web (autores de páginas e criadores de sítios) e aos programadores de ferramentas para criação de conteúdo. Um dos principais documentos nessa área é a WCAG, atualmente em sua versão 2.0, desenvolvida pelo consórcio W3C a partir da criação do WAI (Web Accessibility Initiative), contendo as recomendações de acessibilidade para conteúdo Web. Em nível nacional, o e-MAG é o documento que contém as diretrizes ou recomendações que norteiam o desenvolvimento de sítios e portais acessíveis.

TERCEIRO PASSO: AVALIAÇÃO DE ACESSIBILIDADE

Após a construção do ambiente online de acordo com os padrões Web e as diretrizes de acessibilidade, é necessário testá-lo para garantir sua acessibilidade. No caso dos padrões Web, há um validador automático disponibilizado pelo próprio W3C (ver seção de Recursos). No que diz respeito às diretrizes de acessibilidade, é necessário realizar, inicialmente, uma validação automática através de validadores, que são softwares ou serviços online que ajudam a determinar se um sítio respeitou ou não as recomendações de acessibilidade, gerando um relatório de erros. Uma das ferramentas que podem ser utilizadas é o ASES, avaliador e simulador de acessibilidade em sítios, cujos instrumentos permitem avaliar, simular e corrigir a acessibilidade de páginas, sítios e portais, viabilizando a adoção da acessibilidade por órgãos do governo. Além do ASES, existem outros validadores automáticos (para mais informações, ver seção de Recursos deste documento).

É preciso salientar que, apesar de tornarem a avaliação de acessibilidade mais rápida e menos trabalhosa, os validadores automáticos por si só não determinam se um sítio está ou não acessível. Para uma avaliação efetiva, será necessária uma posterior validação manual.

A validação manual é necessária porque nem todos os problemas de acessibilidade em um sítio são detectados mecanicamente pelos validadores. Para a validação manual, são utilizados checklists de validação humana.

Assim, os passos sugeridos para a avaliação de acessibilidade em um sítio são os seguintes:

1. Validar os códigos do conteúdo HTML e das folhas de estilo;

2. Verificar o fluxo de leitura da página – para tal, utilizar um navegador textual, como o [Lynx](#), ou um leitor de tela (recomendamos o NVDA ou ORCA). Para maiores detalhes, ver documento Descrição dos Leitores de Tela, disponível em: <http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG/material-de-apoio>.
3. Verificar o fluxo de leitura da página sem estilos, sem script e sem as imagens;
4. Verificar as funcionalidades da barra de acessibilidade, aumentando e diminuindo a letra, modificando o contraste, etc.;
5. Realizar a validação automática de acessibilidade utilizando o ASES e outros avaliadores automáticos sugeridos no Capítulo 4;
6. Realizar a **validação manual**, utilizando os checklists de validação humana disponíveis em <http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG/material-de-apoio>.

A validação manual é uma etapa essencial na avaliação de acessibilidade de um sítio, já que os validadores automáticos não são capazes de detectar todos os problemas de acessibilidade em um sítio, pois muitos aspectos requerem um julgamento humano. Por exemplo, validadores automáticos conseguem detectar se o atributo para descrever imagens foi utilizado em todas as imagens do sítio, mas somente uma pessoa poderá verificar se a descrição da imagem está adequada ao seu conteúdo. Para realizar uma validação manual efetiva, o desenvolvedor deverá ter conhecimento sobre as diferentes tecnologias, as barreiras de acessibilidade enfrentadas por pessoas com deficiência e as técnicas ou recomendações de acessibilidade.

Outra etapa essencial da validação de uma página é a realização de testes com usuários reais (pessoas com deficiência ou limitações técnicas). Um usuário real poderá dizer se um sítio está realmente acessível, compreensível e com boa usabilidade e não simplesmente tecnicamente acessível. Quanto maior e mais diversificado o número de usuários reais participando da avaliação de acessibilidade, mais eficaz e robusto será o resultado.

2. Recomendações de acessibilidade

Os padrões de acessibilidade compreendem recomendações ou diretrizes que visam 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 destes padrões também facilita o acesso ao conteúdo da Web, independente 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.

As recomendações de acessibilidade deste documento não estão divididas por níveis de prioridade, já que todas elas são de grande importância e devem ser seguidas. Dessa forma, optou-se por classificar as recomendações nas seguintes seções:

- Marcação
- Comportamento (DOM)
- Conteúdo/Informação
- Apresentação/Design
- Multimídia
- Formulário

OBS: As recomendações deste documento são baseadas em HTML 4.0 e XHTML 1.1. A maioria dos exemplos apresentados nas recomendações a seguir mostram o código (X)HTML que deve ser renderizado no navegador, já que é esse código que apresenta importância para garantir a acessibilidade. Assim, não foram apresentados exemplos do lado do servidor, pois o desenvolvedor pode utilizar a linguagem do lado do servidor que preferir, apenas tomando o cuidado com o código que será gerado.

2.1 Marcação

RECOMENDAÇÃO 1 – RESPEITAR OS PADRÕES DE DESENVOLVIMENTO WEB

Ver WCAG 2.0 Critérios de Sucesso 4.1.1 e 4.1.2

É essencial seguir os padrões de desenvolvimento Web, do W3C (World Wide Web Consortium), com o intuito de maximizar a compatibilidade com atuais e futuros agentes de usuário.

É preciso declarar o DOCTYPE correto da página de qualquer documento HTML ou XHTML. O DOCTYPE define qual versão do (X)HTML o documento está usando e esta é uma informação fundamental para que os agentes de usuário processem corretamente o documento. Além disso, é por meio do DOCTYPE que as ferramentas de validação analisam o código da página e indicam correções. Poderá ser utilizado qualquer DOCTYPE para HTML ou XHTML, com exceção do Frameset. Além disso, qualquer código HTML ou CSS inserido em uma página por script ou outro método similar deve produzir uma página válida quando renderizada.

As camadas lógicas deverão ser separadas, de acordo com o objetivo para o qual elas foram desenvolvidas. Assim, para a camada de conteúdo devem ser utilizadas as linguagens de marcação, como html e xhtml. Para a camada de apresentação visual do conteúdo, utilizam-se as folhas de estilo css em qualquer uma de suas versões. Já para a camada que modifica o comportamento dos elementos, são utilizadas linguagens javascript e modelos de objeto (dom).

Exemplos de DOCTYPE

Em HTML 4.01 Strict

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html lang="pt-BR">
<head>
  <title>Exemplo de DOCTYPE em HTML 4.01</title>
  <meta http-equiv="content-type" content="text/html; charset=utf-8" />
</head>
```

Em XHTML 1.1

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pt-BR">
<head>
```

```
<title> Exemplo de DOCTYPE em XHTML 1.1 </title>
<meta http-equiv="content-type" content="application/xhtml+xml; charset=utf-8" />
</head>
```

Para mais detalhes a respeito dos padrões de desenvolvimento web, ver a *Cartilha de Codificação Padrões Web e-GOV* do padrão e-PWG, disponível em:

<http://www.governoeletronico.gov.br/acoes-e-projetos/padroes-brasil-e-gov>

RECOMENDAÇÃO 2 – ORGANIZAR O CÓDIGO HTML DE FORMA LÓGICA E SEMÂNTICA

Ver WCAG 2.0 Critério de Sucesso 1.3.1

O código HTML deve ser organizado de forma lógica e semântica, ou seja, apresentando os elementos em uma ordem compreensível e correspondendo ao conteúdo desejado. Assim, marcação semântica adequada deve ser utilizada para designar os cabeçalhos (h1, h2, h3), as listas (ul, ol, dl), texto enfatizado (strong), marcação de código (code), marcação de abreviaturas (abbr), marcação de citações longas (blockquote), etc. Dessa forma, as páginas poderão ser apresentadas e compreendidas sem recursos de estilização, tal como as folhas de estilo. Além disso, o código semanticamente correto é muito importante para usuários com deficiência visual, pois os leitores de telas descrevem primeiro o tipo de elemento e depois realizam a leitura do conteúdo que está dentro desse elemento.

Exemplo correto

Padrões Web

- [Menu 1](#)
- [Menu 2](#)

Web Semântica

O poder da web está em sua universalidade. Ser acessada por todos, independente de deficiência, é um aspecto essencial.

Tim Berners Lee

```
<h1>Padrões Web</h1>
<ul>
  <li><a href="menu1.html">Menu 1</a></li>
  <li><a href="menu2.html">Menu 2</a></li>
</ul>
<h2>Web Semântica</h2>
<blockquote>
  O poder da web está em sua universalidade. Ser
  acessada por todos, independente de deficiência, é um
  aspecto essencial.
</blockquote>
<cite xml:lang="en">Tim Berners Lee</cite>
```

Exemplo Incorreto**Padrões Web**[Menu 1](#)[Menu 2](#)**Web Semântica**

O poder da web está em sua universalidade. Ser acessada por todos, independente de deficiência, é um aspecto essencial

Tim Berners Lee

```
<h1>Padrões Web</h1>
<p><a href="menu1.html">Menu 1</a></p>
<p><a href="menu2.html">Menu 2</a></p>
```

```
<h2>Web Semântica</h2>
```

```
<p>
```

O poder da web está em sua universalidade. Ser acessada por todos, independente de deficiência, é um aspecto essencial.

```
</p>
```

```
<p>Tim Berners Lee</p>
```

RECOMENDAÇÃO 3 – UTILIZAR CORRETAMENTE OS NÍVEIS DE CABEÇALHO

Ver WCAG 2.0 Critérios de Sucesso 1.3.1 e 2.4.10

Os níveis de cabeçalho devem ser utilizados de forma lógica e semântica para facilitar a leitura e compreensão. Além disso, pessoas acessando uma página com leitor de tela podem navegar através dos cabeçalhos, pulando de um para outro, agilizando, assim, a navegação. Conceitualmente, existem seis níveis de títulos, sendo o h1 o mais alto, ou seja, deverá corresponder ao título principal da página. Dessa forma, cada página deverá ter apenas um h1, o qual poderá ser substituído por uma imagem, mas deverá permanecer com seu conteúdo, mesmo que não visualmente, permitindo a leitura pelo leitor de tela. Já os níveis do h2 ao h6 poderão ser utilizados mais de uma vez na página, mas sem excesso e com lógica textual. Para compreender melhor os níveis de título pode-se tomar como exemplo um sítio de um livro, onde o nome do livro é o h1, os capítulos são h2, os subcapítulos são h3 e assim por diante.

Exemplo

Técnicas culinárias

A seguir os segredos que facilitam a vida na cozinha.

Legumes, folhas e vegetais

Baba do quiabo

Para eliminar a baba do quiabo, lave-o ainda inteiro, seque-o e coloque-o numa tigela com um pouco de suco de limão, deixando repousar durante 15 minutos. Depois lave ligeiramente, corte e cozinhe.

Feijão

1 xícara de feijão cru serve três pessoas depois de pronto.

Cenouras e aipos

Para resolver o problema de cenouras e aipos meio murchos, mergulhe-os em água gelada misturada com uma colher de chá de mel por uma hora. Escorra e seque levemente depois.

Congelamento e descongelamento

Carne em pedaços

Para descongelar carne em pedaços inteiros coloque-a embrulhada, numa vasilha com água. Coloque sal na água e no pacote e tampe por uma hora.

Carne moída

Para apressar o descongelamento da carne moída, salgue a quantidade que irá usar. O sal apressa o descongelamento.

Figura 1 – Exemplo de utilização dos níveis de título

HTML

```
<h1>Técnicas culinárias</h1>
<p>A seguir os segredos que facilitam a vida na cozinha.</p>

<h2>Legumes, folhas e vegetais</h2>
<h3>Baba do quiabo</h3>

<p>Para eliminar a baba do quiabo, lave-o ainda inteiro, seque-o e coloque-o numa tigela com um pouco de suco de limão, deixando repousar durante 15 minutos. Depois lave ligeiramente, corte e cozinhe.</p>

<h3>Feijão</h3>
<p>1 xícara de feijão cru serve três pessoas depois de pronto.</p>

<h3>Cenouras e aipos</h3>
<p>Para resolver o problema de cenouras e aipos meio murchos, mergulhe-os em água gelada misturada com uma colher de chá de mel por uma hora. Escorra e seque levemente depois.</p>
```



```
<h2>Congelamento e descongelamento</h2>
```

```
<h3>Carne em pedaços</h3>
```

```
<p>Para descongelar carne em pedaços inteiros coloque-a embrulhada, numa vasilha com água. Coloque sal na água e no pacote e tampe por uma hora.</p>
```

```
<h3>Carne moída</h3>
```

```
<p>Para apressar o descongelamento da carne moída, salgue a quantidade que irá usar. O sal apressa o descongelamento.</p>
```

RECOMENDAÇÃO 4 – ORDENAR DE FORMA LÓGICA E INTUITIVA A LEITURA E TABULAÇÃO

Ver WCAG 2.0 Critérios de Sucesso 1.3.2 e 2.4.3

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 disponibilizar o bloco de conteúdo no HTML antes do bloco de menu, para que usuários, navegando pelo teclado, não precisem navegar por todos os itens de menu antes de chegar ao conteúdo. Apesar de os atalhos auxiliarem nesse sentido, alguns usuários não sabem utilizá-los. Os atalhos não funcionam em interfaces especializadas, como o do Leitor de Tela DOSVOX e podem ser de difícil utilização para pessoas com deficiência motora.

OBS: O atributo `tabindex` somente deverá ser utilizado quando existir real necessidade. Com o uso de CSS para fins de leiaute, o código HTML pode facilmente ser desenvolvido de maneira que a ordem de tabulação seja a correta. No entanto, se houver necessidade de utilizar o `tabindex`, o mesmo deverá ser utilizado com a semântica correta e deverá ser verificado manualmente se o fluxo fornecido pelo `tabindex` é realmente o desejado, evitando, assim, que o uso do `tabindex` resulte em uma ordem de tabulação inconsistente.

Exemplo correto (sem o uso do `tabindex`)

```
<ul>
  <li><a href="#">Página Inicial</a></li> <!--primeiro foco -->
  <li><a href="#">Capítulo 1</a></li> <!--segundo foco -->
  <li><a href="#">Capítulo 2</a></li> <!--terceiro foco -->
  <li><a href="#">Capítulo 3</a></li> <!--quarto foco -->
</ul>
```

Exemplo incorreto do uso do tabindex

```
<ul>
<li><a href="main.html" tabindex="1">Página Inicial</a></li>
<li><a href="capitulo1.html" tabindex="4">Capítulo 1</a></li>
<li><a href="capitulo2.html" tabindex="3">Capítulo 2</a></li>
<li><a href="capitulo3.html" tabindex="2">Capítulo 3</a></li>
</ul>
```

Ordem de tabulação errada, conferida pelo tabindex:

- | | | | |
|----------------------------------|----------------------------------|----------------------------------|----------------------------------|
| • Página Inicial | • Página Inicial | • Página Inicial | • Página Inicial |
| • Capítulo 1 | • Capítulo 1 | • Capítulo 1 | • Capítulo 1 |
| • Capítulo 2 | • Capítulo 2 | • Capítulo 2 | • Capítulo 2 |
| • Capítulo 3 | • Capítulo 3 | • Capítulo 3 | • Capítulo 3 |

RECOMENDAÇÃO 5 - DISPONIBILIZAR TODAS AS FUNÇÕES DA PÁGINA VIA TECLADO

Ver WCAG 2.0 Critérios de Sucesso 2.1.1 e 2.1.2

Todas as funções da página desenvolvidas utilizando-se linguagens de script (javascript) deverão estar disponíveis quando for utilizado apenas o teclado. É importante salientar que o foco não deverá estar bloqueado ou fixado em um elemento da página, para que o usuário possa mover-se pelo teclado por todos os elementos.

Algumas funções específicas do mouse possuem uma função lógica correspondente via teclado, conforme mostrado na tabela a seguir:

Evento do mouse	Evento do teclado
onmousedown	onkeydown
onmouseup	onkeyup
onclick*	onkeypress
onmouseover	onfocus*
onmouseout	onblur*

OBS: * Alguns manipuladores de eventos são dispositivo-independentes, ou seja, se aplicam a qualquer dispositivo (mouse, teclado ou outro), como é o caso de: onFocus, onBlur, onSelect, onChange, e onClick (quando o onClick for utilizado em um link ou elemento de formulário).

Quando forem utilizados múltiplos manipuladores de eventos para uma ação, de maneira que ela funcione tanto pelo mouse como pelo teclado, é importante testar o resultado final em diferentes navegadores e utilizando diferentes recursos de tecnologia assistiva, para garantir que o evento seja, de fato, acessível.

Dê preferência por utilizar o onclick/onkeypress em vez de onmousedown/onkeydown e onmouseup/onkeyup, pois estes últimos fazem com que o evento seja disparado automaticamente através do teclado. Se houver real necessidade de utilização destes eventos, deverá ser feito um controle sobre qual tecla deverá ser acionada para que o evento ocorra, conforme o exemplo a seguir:

JavaScript

```
<script type="text/javascript">
var x=document.getElementById("link")
x.onkeydown=function(e){
  var pressedkey
  if(typeof event!='undefined'){ //navegador Internet Explorer
    pressedkey=window.event.keyCode
  }else{//outros navegadores
    pressedkey=e.keyCode //identifica tecla pressionada
  }
  if(pressedkey=='13'){ //teste se a tecla é o "enter"
    window.open('http://www.brasil.gov.br/') //abre a URL
  }
}
</script>
```

HTML

```
<p><a href="#" id="link">Portal Brasil</a></p>
```

Existem funções do mouse que não possuem uma função correspondente via teclado, como é o caso de duplo clique (dblclick). Nesses casos, é necessário implementar a função de maneira alternativa, como, por exemplo, incluindo botões que executem, pelo teclado, a função de forma equivalente. O evento onclick já funciona pelo teclado (tecla ENTER) na maioria dos navegadores.

RECOMENDAÇÃO 6 – FORNECER ÂNCORAS PARA IR DIRETO A UM BLOCO DE CONTEÚDO

Ver WCAG 2.0 Critério de Sucesso 2.4.1

Devem ser fornecidas âncoras, disponíveis na barra de acessibilidade, que apontem para links relevantes presentes na mesma página. Assim, é possível ir ao bloco de

conteúdo desejado. Os links devem ser colocados em lugares estratégicos da página, como por exemplo, no início e final do menu, do conteúdo, etc.

Para facilitar a utilização das âncoras, podem ser disponibilizados atalhos por teclado, utilizando o atributo `accesskey` nos links relevantes. É importante ressaltar que as dicas de atalhos presentes na barra de acessibilidade não devem possuir o atributo `accesskey`, já que não pode haver repetição do mesmo `accesskey` em uma página. Recomenda-se fornecer atalhos para o menu principal, para o conteúdo e para a caixa de pesquisa.

Exemplo

Topo da Página (na barra de acessibilidade)

```
<ul id="atalhos">
  <li><a href="#conteudo">Ir para conteúdo [1]</a></li>
  <li><a href="#menu">Ir para menu principal[2]</a></li>
  <li><a href="#busca">Ir para busca [3]</a></li>
</ul>
```

Conteúdo da Página

```
<div>
  <a name="conteudo" id="conteudo" class="oculto" accesskey="1">Início do
  conteúdo</a>
  <!-- Conteúdo →
</div>
```

Menu Principal da Página

```
<div>
  <a name="menu" id="menu" class="oculto" accesskey="2">Início do menu</a>
  <!--itens de menu -->
</div>
```

Formulário de pesquisa do sítio (pode estar em qualquer lugar no sítio)

```
<form action="#" method="post">
  <fieldset>
    <legend>Buscar</legend>
    <label for="busca">Pesquise aqui</label>
    <input type="text" id="busca" name="busca" accesskey="3" value="Pesquise
    aqui" />
    <input type="submit" value="Buscar" class="buscar" name="buscar" />
  </fieldset>
</form>
```

Foram utilizados ambos os atributos name e id para que as âncoras funcionem em todos os navegadores, tanto textuais quanto gráficos, já que há os que suportam ambos os atributos e os que suportam apenas um deles.

As dicas dos principais atalhos do sítio devem ser disponibilizadas na barra de acessibilidade e na página sobre a acessibilidade do sítio. Maiores detalhes a esse respeito podem ser encontrados no *capítulo 3 – Elementos de acessibilidade nas páginas do Governo Federal*.

RECOMENDAÇÃO 7 – NÃO UTILIZAR TABELAS PARA DIAGRAMAÇÃO

Ver WCAG 2.0 Critério de Sucesso 1.3.1 (Técnica H51)

As tabelas devem ser utilizadas apenas para dados tabulares e não para efeitos de disposição dos elementos na página. Para este fim, utilize as folhas de estilo.

RECOMENDAÇÃO 8 – SEPARAR LINKS ADJACENTES

Ver WCAG 2.0 Critério de Sucesso 1.3.1 (Técnica H48)

Links adjacentes devem ser separados por mais do que simples espaços, para que não fiquem confusos, em especial para usuários que utilizam leitor de tela. Para isso, é recomendado o uso de listas, onde 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 um ao lado do outro, por exemplo.

Caso os links estejam no meio de um parágrafo, pode-se utilizar vírgulas, parênteses, colchetes, etc., para fazer a separação.

Exemplo correto

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

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="mapasite.html">Mapa do Site</a>  
</p>  
<!-- Conteúdo do Site -->
```

RECOMENDAÇÃO 9 – NÃO ABRIR NOVAS INSTÂNCIAS SEM A SOLICITAÇÃO DO USUÁRIO

Ver WCAG 2.0 Critério de Sucesso 3.2.5

A decisão de se utilizar-se de novas instâncias – por exemplo abas ou janelas - para acesso a páginas e serviços ou qualquer informação é do cidadão. Assim, não devem ser utilizados:

- Pop-ups;
- A abertura de novas abas ou janelas;
- O uso do atributo target="_blank";
- Mudanças no controle do foco do teclado;
- Entre outras, que não tenham sido solicitadas pelo usuário.

2.2 Comportamento (DOM)

RECOMENDAÇÃO 10 – GARANTIR QUE OS OBJETOS PROGRAMÁVEIS SEJAM ACESSÍVEIS

Ver WCAG 2.0 Critérios de Sucesso 2.1.1 e 2.1.2

Deve-se garantir que scripts, Flash, conteúdos dinâmicos e outros elementos programáveis sejam acessíveis. Se não for possível que o elemento programável seja diretamente acessível, deve ser fornecida uma alternativa em HTML para o conteúdo. Assim, é preciso garantir que o conteúdo e as funcionalidades de objetos programáveis sejam acessíveis aos recursos de tecnologia assistiva e que seja possível a navegação por teclado.

Quando o script for utilizado em uma página da Web, uma forma de fornecer uma alternativa para ele é através do elemento <noscript>. Este 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.

Exemplo correto

Página HTML

```
<a href="cadastro.html" id="cadastro">Cadastre-se agora!</a>
```

Página JavaScript (.js)

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

Exemplo incorreto

Página HTML

```
<a href="javascript:pop()">Cadastre-se agora!</a>  
<script language="javascript" type="text/javascript">  
    function pop() {  
        alert("Você vai fazer um novo cadastro!");  
    }  
</script>
```

Nesse caso, se o navegador não tiver suporte a scripts, o usuário ficará impossibilitado de acessar o link.

OBS: A função "alert" do javascript não gera um pop-up e sim uma mensagem que é lida por todos os leitores de tela.

RECOMENDAÇÃO 11- NÃO CRIAR PÁGINAS COM ATUALIZAÇÃO AUTOMÁTICA PERIÓDICA

Ver WCAG 2.0 Critério de Sucesso 3.2.5 (Técnicas SVR1 e H76)

Não devem ser criadas páginas com atualização automática periódica. Assim, não deve ser utilizada a meta tag refresh, nem outra forma de atualização automática. Páginas que se atualizam automaticamente podem confundir e desorientar os usuários, especialmente usuários que utilizam leitores de tela.

Exemplo1

Em uma interface Web para e-mail (Webmail), um desenvolvedor pode fornecer um botão ou link para buscar novos e-mails recebidos em vez de atualizar automaticamente.

Em páginas onde o limite de tempo é absolutamente necessário, o usuário deverá ser informado que a página é atualizada automaticamente.

Exemplo2

Em uma página de leilões online, é fornecido um banner contendo a informação de que a página é atualizada a cada 15 segundos.

RECOMENDAÇÃO 12 – NÃO UTILIZAR REDIRECIONAMENTO AUTOMÁTICO DE PÁGINAS

Ver WCAG 2.0 Critério de Sucesso 3.2.5 (Técnicas SVR1 e H76)

Não devem ser utilizadas marcações para redirecionar para uma nova página, como a meta tag *refresh*. Ao invés disso, deve-se configurar o servidor para que o redirecionamento seja transparente para o usuário.

Exemplo Incorreto

```
<head>
  <title>Exemplo</title>
  <meta http-equiv="refresh" content="20; url='http://www.exemplo.com/'" />
</head>
<body>
  <p>Esta página mudou seu endereço para www.novoendereco.com.br</p>
</body>
```

RECOMENDAÇÃO 13 – FORNECER ALTERNATIVA PARA MODIFICAR LIMITE DE TEMPO

Ver WCAG 2.0 Critério de Sucesso 2.2.1

Em uma página onde há limite de tempo para realizar uma tarefa deve haver a opção de desligar, ajustar ou prolongar esse limite.

Essa recomendação não se aplica a eventos em que o limite de tempo é absolutamente necessário.

Exemplo : Na inserção de dados em um formulário para obtenção de um benefício ou consulta a processo, o cidadão deve ter tanto tempo quanto for necessário para o preenchimento de seus dados.

RECOMENDAÇÃO 14 – NÃO INCLUIR SITUAÇÕES COM INTERMITÊNCIA DE TELA

Ver WCAG 2.0 Critério de Sucesso 2.3.1

Não devem ser utilizados efeitos visuais piscantes, intermitentes ou cintilantes. Em pessoas com epilepsia fotosensitiva, o cintilar ou piscar pode desencadear um ataque epilético. A exigência dessa diretriz aplica-se também para propaganda de terceiros inserida na página.

RECOMENDAÇÃO 15 – ASSEGURAR O CONTROLE DO USUÁRIO SOBRE AS ALTERAÇÕES TEMPORAIS DO CONTEÚDO

Ver WCAG 2.0 Critério de Sucesso 2.2.2

Conteúdos que “se movem”, rolagens, movimentações em geral ou animações não devem ser disparadas automaticamente sem o controle do usuário, mesmo em propagandas na página. Ao usuário deve ser repassado o controle sobre essas movimentações (quer seja por escolha de preferência de visualização da página, quer por outro método qualquer acessível a usuário com deficiência). Além disso, o usuário deve ser capaz de parar e reiniciar conteúdos que se movem, sem exceção.

A velocidade desses conteúdos também deve ser passível de controle pelo usuário, a menos que a implementação de mecanismo para alterar a velocidade seja uma tarefa de difícil execução (se for necessário realizar uma escolha baseando-se nas limitações, prefira implementar mecanismos para reduzir a velocidade dos conteúdos no lugar de aumentar).

2.3 Conteúdo / Informação

RECOMENDAÇÃO 16 – IDENTIFICAR O IDIOMA PRINCIPAL DA PÁGINA

Ver WCAG 2.0 Critério de Sucesso 3.1.1

Deve-se identificar o principal idioma utilizado nos documentos. A identificação é feita por meio do atributo lang do HTML e, para documentos XHTML, é utilizado o xml:lang.

Exemplo

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">  
<head>  
  <title>documento escrito em português do Brasil</title>  
...
```

RECOMENDAÇÃO 17 – OFERECER UM TÍTULO DESCRITIVO E INFORMATIVO À PÁGINA

Ver WCAG 2.0 Critério de Sucesso 2.4.2

O título da página deve ser descritivo e informativo, já que essa informação será a primeira lida pelo leitor de tela, quando o usuário acessar a página. O título é informado pela tag <title>.

Exemplo

O sítio sobre o Projeto Acessibilidade Virtual da RENAPI (Rede Nacional de Pesquisa e Inovação em Tecnologias Digitais) apresenta o seguinte título:

```
<title>  
  Projeto Acessibilidade Virtual - Portal RENAPI – Página Inicial  
</title>
```

RECOMENDAÇÃO 18 – DISPONIBILIZAR INFORMAÇÃO SOBRE A LOCALIZAÇÃO DO USUÁRIO NA PÁGINA

Ver WCAG 2.0 Critério de Sucesso 2.4.8

Deverá ser fornecido um mecanismo que permita ao usuário orientar-se dentro de um conjunto de páginas, permitindo que ele saiba onde está no momento. Assim, poderá ser utilizado o recurso de “migalha de pão” (*breadcrumbs*), que são links navegáveis em forma de lista hierárquica e permitem que o usuário saiba qual o caminho percorrido até chegar à página em que se encontra no momento.

Exemplo

Um usuário navegando por um sítio de uma universidade encontra-se na seção de editais, que está dentro do menu “Ensino”. Acima do conteúdo, é disponibilizado a seguinte Migalha de pão:

Você está em: [Página inicial](#) > [Ensino](#) > **Editais**

OBS: Na migalha de pão, todas as páginas do caminho, com exceção da qual está o usuário (posição atual), deverão estar implementadas como links.

RECOMENDAÇÃO 19 – DESCREVER LINKS CLARA E SUCINTAMENTE

Ver WCAG 2.0 Critérios de Sucesso 2.4.4 e 2.4.9

Deve-se identificar claramente o destino de cada link, informando, inclusive, se o link remete a outro sítio. Além disso, é preciso que o texto do link faça sentido mesmo quando isolado do contexto da página.

É preciso tomar cuidado para não utilizar o mesmo título para dois ou mais links que apontem para destinos diferentes.

Exemplo 1

```
<h2>Educação Superior</h2>
<p>Tomam posse os reitores das federais da Bahia e Triângulo</p>

<p> <a href="notici5125.html">Leia mais notícias sobre Educação Superior</a>
</p>
```

Repare que não há necessidade de utilizar o atributo title em links, já que esse atributo não é bem suportado por recursos de tecnologia assistiva, como leitores de tela, não tem utilidade para quem navega apenas pelo teclado e não tem bom suporte em dispositivos móveis, como celulares, entre outros problemas. Assim, se você quiser fornecer informações adicionais para um link, faça-o no próprio texto do link ou no contexto.

Exemplo 2

Correto:

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

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

Incorreto:

[Clique aqui](#) para ganhar um prêmio fornecido pelo nosso patrocinador

OBS: Não é recomendada a utilização de links do tipo “clique aqui” pois esta expressão não faz sentido fora do contexto. Muitos usuários de leitores de tela navegam por links, tornando descrições como “Clique aqui”, “Veja mais” insuficientes para o usuário saber o destino do link, ou localizá-lo na página. Evitar essas expressões para evitar verborragia, ou seja, evitar que o leitor de tela “leia” para o usuário palavras ou frases desnecessárias.

RECOMENDAÇÃO 20 – FORNECER ALTERNATIVA EM TEXTO PARA AS IMAGENS DO SÍTIO

Ver WCAG 2.0 Critério de Sucesso 1.1.1 (Técnica G95)

Deve ser fornecida uma descrição para as imagens da página, utilizando-se o atributo alt. Imagens que não transmitem conteúdo, ou seja, imagens decorativas, devem ser inseridas por CSS.

No entanto, descrever qualquer imagem, em geral, é algo bastante subjetivo e a descrição deve ser adaptada ao contexto em que a imagem se encontra. Para maiores detalhes de como escrever um texto alternativo veja o tutorial *O uso correto do texto alternativo* na seção do e-MAG no portal de Governo Eletrônico.

Exemplo 1



Figura 2 – Exemplo de descrição de imagem. Foto de Jônatas Cunha em licença Creative Commons Fonte: <http://www.flickr.com/photos/jonycunha/4019704214/in/photostream/>

No código:

```

```

Exemplo 2

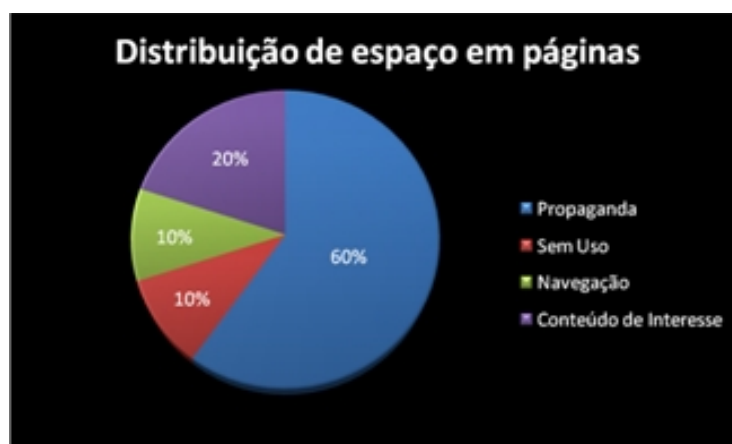


Figura 3 – Exemplo de descrição de banner

No código:

```
<a href="http://www.dominiopublico.gov.br/">
  
</a>
```

Apesar de não haver um limite de caracteres no atributo alt, ele é utilizado para descrições sintéticas, em poucas palavras ou em uma frase curta. Para imagens mais complexas que exigem uma descrição mais detalhada, como gráficos, por exemplo, deve-se fornecer, além do alt, a descrição no próprio contexto ou um link para a descrição longa logo após a imagem. Deve ficar claro para o usuário que esse link remete para a descrição longa da imagem, conforme o exemplo a seguir.

Exemplo 3**Figura 4 – Exemplo de descrição de gráfico**

[A descrição textual do gráfico](#) está disponível em outra página

No código:

```


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

Para gráficos, poderá ser disponibilizada tanto a descrição textual quanto a tabela de dados que lhe deu origem, desde que a tabela seja fornecida em HTML, tomando-se os devidos cuidados relativos à acessibilidade.

Além de fornecer a descrição longa no contexto ou em um link próximo, pode-se utilizar o atributo longdesc na imagem, o qual é recomendado pela WCAG para descrições longas, mas não é suportado por alguns recursos de tecnologia assistiva. A utilização do longdesc pode ser vista no exemplo a seguir.

Exemplo 4

```


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

graficoPesquisa.html

```
<h2> Distribuição de espaço em páginas</h2>
<ul>
  <li>Propaganda - 60%</li>
  <li>Sem Uso - 10%</li>
  <li>Navegação - 10%</li>
```

```
<li>Conteúdo de Interesse - 20%</li>
</ul>
```

Exemplo de imagem decorativa:

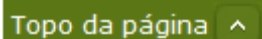


Figura 5 – Imagem decorativa em link “Volta ao topo da página”

HTML

```
<a href="#id_de_destino">Topo da página</a>
```

CSS

```
a{
background:transparent url(img/topo_pagina.jpg)no-repeat right top; /*inserção da imagem decorativa
do link */
padding:4px 27px 0 0; /* espaço colocado para a imagem ficar ao lado do link */
height:23px; /*ajuste na altura do link para que apareça toda a imagem */
}
```

RECOMENDAÇÃO 21 – FORNECER ALTERNATIVA EM TEXTO PARA AS ZONAS ATIVAS DE MAPA DE IMAGEM

Ver WCAG 2.0 Critério de Sucesso 1.1.1 (Técnica H24)

Para mapas de imagem do lado do cliente, devem ser fornecidas descrições através do atributo alt para cada uma das zonas ativas, ou seja, para cada um dos links que receberá o foco.

Exemplo (mapa de imagem do lado do cliente)



Figura 6 – Mapa de imagem exemplo 1

```

<map name="Map" id="Map">
  <area shape="rect" coords="8,10,63,59" href="a.html" alt="Link para a seção [A]"
/>
  <area shape="rect" coords="77,9,126,61" href="b.html" alt="Link para a seção [B]"
/>
</map>
```

Além dos mapas de imagem do lado do cliente, existem os do lado do servidor. No

entanto, é recomendada a utilização de mapas de imagem do lado do cliente, já que para mapas de imagem do lado do servidor não é possível fornecer um alt para cada uma das zonas ativas, somente para o mapa como um todo, não sendo possível, portanto, torná-lo acessível. No entanto, se for realmente necessária sua utilização, devem ser fornecidos links redundantes relativos a cada região ativa do mapa de imagem, conforme o exemplo a seguir, para que, desta forma, usuários com leitores de tela possam ter acesso ao seu conteúdo.

Exemplo (mapa de imagem do lado do servidor)



[Área Verde](#) - [Área Amarela](#) - [Área Azul](#)

```
<a href="novaPagina.jpg"></a>
<p><a href="areaVerde.html">Área Verde</a> -
</p>
<p><a href="areaAmarela.html">Área
Amarela</a> - </p>
<p><a href="areaAzul.html">Área Azul</a></p>
```

Figura 7 – Mapa de imagem exemplo 2

OBS: Não devem ser utilizados mapas de imagem para menus ou seleção de regiões para serviços.

RECOMENDAÇÃO 22 – DISPONIBILIZAR DOCUMENTOS EM FORMATOS ACESSÍVEIS

Sem correspondência a critérios da WCAG 2.0

Os documentos devem ser disponibilizados preferencialmente em HTML. Também podem ser utilizados arquivos para download no formato ODF, tomando-se os cuidados para que sejam acessíveis. Se um arquivo for disponibilizado em PDF, deverá ser fornecida uma alternativa em HTML ou ODF. É necessário, também, informar a extensão e o tamanho do arquivo no próprio texto do link.

Exemplo

[Manual do W3C \(formato .odt, tamanho 150Kb\)](#)

OBS: O ODF (Open Document Format) é um formato aberto de documento adotado pela e-PING (Arquitetura de Interoperabilidade em Governo Eletrônico) que pode ser implementado em qualquer sistema. O ODF engloba formatos como: ODT para documentos de texto, ODS para planilhas eletrônicas, ODP para apresentações de slides, entre outros.

Muitos softwares já utilizam esses formatos, como é o caso do OpenOffice, BrOffice, Google Docs, Abiword e StarOffice. O Microsoft Office 2010 também inclui suporte para ODF. Para versões anteriores do Microsoft Office, pode ser instalado um Add-in gratuito para suporte aos formatos ODF.

RECOMENDAÇÃO 23 – EM TABELAS, UTILIZAR TÍTULOS E RESUMOS DE FORMA

APROPRIADA

Ver WCAG 2.0 Critério de Sucesso 1.3.1 (Técnicas H39 e H73)

O título da tabela deve ser definido pelo elemento caption e deve ser o primeiro elemento utilizado após a declaração do elemento table. Em casos de tabelas extensas, deve ser fornecido um resumo de seus dados através do atributo summary que deve ser declarado no elemento table.

Exemplo

```
<table summary="Esta tabela exhibe os copos de café consumidos por cada senador, o tipo de  
café (descafeinado ou normal), com açúcar ou sem açúcar.">  
  <caption>Copos de café por Senador</caption>  
  ...
```

Para mais detalhes veja o tutorial [tabelas acessíveis](#) (documento pdf - 154 KB), na seção do e-MAG no Portal do Programa de Governo Eletrônico.

RECOMENDAÇÃO 24 – ASSOCIAR CÉLULAS DE DADOS ÀS CÉLULAS DE CABEÇALHO EM UMA TABELA

Ver WCAG 2.0 Critério de Sucesso 1.3.1 (Técnicas H43 e H63)

Em tabelas de dados simples, a uso apropriado do elemento th para os cabeçalhos e do elemento td para as células de dados é essencial para torná-las acessíveis. Para incrementar a acessibilidade, deve-se utilizar os elementos thead, tbody e tfoot, para

agrupar as linhas de cabeçalho, do corpo da tabela e do final, respectivamente, com exceção de quando a tabela possuir apenas o corpo, sem ter seções de cabeçalho e rodapé. O W3C sugere utilizar o tfoot antes do tbody dentro da definição table para que o agente de usuário possa renderizar o rodapé antes de receber todas (potencialmente numerosas) linha de dados.

Exemplo 1

```
<table>
  <caption>Demonstrativo do Patrimônio</caption>
  <thead>
    <tr>
      <th>Tipos</th>
      <th>Valores (R$)</th>
      <th>Percentual</th>
    </tr>
  </thead>
  <tfoot>
    <tr>
      <td>Total</td>
      <td>110.740,22</td>
      <td>100%</td>
    </tr>
  </tfoot>
  <tbody>
    <tr>
      <td>Recursos Financeiro</td>
      <td>56.879,63</td>
      <td>51,36%</td>
    </tr>
    <tr>
      <td>Bens Móveis</td>
      <td>25.691,23</td>
      <td>23,20%</td>
    </tr>
    <tr>
      <td>Bens Imóveis</td>
      <td>28.169,36</td>
      <td>25,44%</td>
    </tr>
  </tbody>
</table>
```

Demonstrativo do Patrimônio		
Tipos	Valores (R\$)	Percentual
Recursos Financeiro	56.879,63	51,36%
Bens Móveis	25.691,23	23,20%
Bens Imóveis	28.169,36	25,44%
Total	110.740,22	100%

Figura 8 – Imagem da tabela descrita no código do exemplo 1

Para tabelas mais complexas, é necessário utilizar marcações para associar as células de dados com as células de cabeçalho. A maneira mais adequada de realizar esse procedimento é utilizar os elementos id/headers ou scope/col. No primeiro, pode-se associar qualquer célula de conteúdo a qualquer célula de cabeçalho, utilizando o mesmo valor para o atributo id e para o headers. No segundo caso, a associação é automática, sendo mais utilizado em tabelas de associação direta, onde é dado o valor "col" para o atributo scope nos cabeçalhos. Nos exemplo a seguir, é possível verificar a utilização do id/headers e do scope/col.

Exemplo 2

`<table summary="...">`

```
<table>
  <caption>Resultado do Concurso</caption>
  <tr>
    <th id="vaga">Vaga</th>
    <th id="candidato">Nome do candidato</th>
    <th id="basico">Prova de Conhecimento Básico</th>
    <th id="especifico">Prova de Conhecimento Específico</th>
  </tr>
  <tr>
    <td id="adm" rowspan="2">Técnico Administrativo</td>
    <td id="PaulodaSilva">Paulo da Silva</td>
    <td headers="adm basico PaulodaSilva">8</td>
    <td headers="adm especifico PaulodaSilva">16</td>
  </tr>
  <tr>
    <td id="PedroPontes">Pedro Pontes</td>
    <td headers="adm basico PedroPontes">7</td>
    <td headers="adm especifico PedroPontes">15</td>
  </tr>
  <tr>
    <td id="inf">Técnico em Informática</td>
    <td id="JoaoPereira">João Pereira</td>
    <td headers="inf basico JoaoPereira">9</td>
    <td headers="inf especifico JoaoPereira">17</td>
  </tr>
</table>
```

Resultado do Concurso

Vaga	Nome do candidato	Prova de Conhecimento Básico	Prova de Conhecimento Específico
Técnico Administrativo	Paulo da Silva	8	16
	Pedro Pontes	7	15
Técnico em Informática	João Pereira	9	17

Figura 9 – Imagem da tabela descrita no código do exemplo 2

Exemplo 3

```

<table summary="Tabela de vendas de maçã e banana efetuadas no ano de 2010">
<caption>Vendas 2010</caption>
  <tr>
    <th scope="col">Mês</th>
    <th scope="colgroup" colspan="2">Rio Grande do Sul</th>
    <th scope="colgroup" colspan="2">Santa Catarina</th>
  </tr>
  <tr>
    <td>Janeiro</td>
    <td scope="col">Maçã</td>
    <td scope="col">Banana</td>
    <td scope="col">Maçã</td>
    <td scope="col">Banana</td>
  </tr>
  <tr>
    <td>Fevereiro</td>
    <td>1000</td>
    <td>1500</td>
    <td>3000</td>
    <td>1000</td>
  </tr>
  <tr>
    <td>Março</td>
    <td>2000</td>
    <td>1500</td>
    <td>3500</td>
    <td>500</td>
  </tr>
</table>

```

Vendas 2010

Mês	Rio Grande do Sul		Santa Catarina	
Janeiro	Maçã	Banana	Maçã	Banana
Feveiro	1000	1500	3000	1000
Março	2000	1500	3500	500

Figura 10 – Imagem da tabela descrita no código do exemplo 3

Exemplo 4

```
<table summary="Tabela com resumo das despesas de transporte durante o mês de
Agosto nas cidades de Porto Alegre e Florianópolis">
  <caption>Relatório de despesa de viagem</caption>
  <tr>
    <th></th>
    <th id="alimentacao" axis="despesas">Alimentação</th>
    <th id="hotel" axis="despesas">Hotel</th>
    <th id="transporte" axis="despesas">Transporte</th>
  </tr>
  <tr>
    <th id="poa" axis="localização" colspan="4">Porto Alegre</th>
  </tr>
  <tr>
    <td id="data1" axis="data">25 de agosto de 2010</td>
    <td headers="poa data1 alimentacao">R$ 37,74</td>
    <td headers="poa data1 hotel">R$ 112,00</td>
    <td headers="poa data1 transporte">R$ 45,00</td>
  </tr>
  <tr>
    <td id="data2" axis="data">26 de agosto de 2010</td>
    <td headers="poa data2 alimentacao">R$ 27,28</td>
    <td headers="poa data2 hotel">R$ 112,00</td>
    <td headers="poa data2 transporte">R$ 45,00</td>
  </tr>
  <tr>
    <th id="subPoa">Subtotal</th>
    <td headers="poa subPoa alimentacao">R$ 65,02</td>
    <td headers="poa subPoa hotel">R$ 224,00</td>
    <td headers="poa subPoa transporte">R$ 90,00</td>
  </tr>
  <tr>
    <th id="floripa" axis="localização" colspan="4">Florianópolis</th>
  </tr>
  <tr>
    <td id="data3" axis="data">27 de agosto de 2010</td>
    <td headers="floripa data3 alimentacao">R$ 96,25</td>
    <td headers="floripa data3 hotel">R$ 109,00</td>
    <td headers="floripa data3 transporte">R$ 36,00</td>
  </tr>
  <tr>
    <td id="data4" axis="data">28 de agosto de 2010</td>
    <td headers="floripa data4 alimentacao">R$ 35,00</td>
    <td headers="floripa data4 hotel">R$ 109,00</td>
    <td headers="floripa data4 transporte">R$ 36,00</td>
  </tr>
</table>
```

```

<th id="subFloripa">Subtotal</th>
<td headers="floripa subFloripa alimentacao">R$ 131,25</td>
<td headers="floripa subFloripa hotel">R$ 218,00</td>
<td headers="floripa subFloripa transporte">R$ 72,00</td>
</tr>
<tr>
<th id="total">Total</th>
<td headers="total alimentacao">R$ 196,27</td>
<td headers="total hotel">R$ 442,00</td>
<td headers="total transporte">R$ 162,00</td>
</tr>
</table>

```

Relatório de despesa de viagem			
	Alimentação	Hotel	Transporte
Porto Alegre			
25 de agosto de 2010	R\$ 37,74	R\$ 112,00	R\$ 45,00
26 de agosto de 2010	R\$ 27,28	R\$ 112,00	R\$ 45,00
Subtotal	R\$ 65,02	R\$ 224,00	R\$ 90,00
Florianópolis			
27 de agosto de 2010	R\$ 96,25	R\$ 109,00	R\$ 36,00
28 de agosto de 2010	R\$ 35,00	R\$ 109,00	R\$ 36,00
Subtotal	R\$ 131,25	R\$ 218,00	R\$ 72,00
Total	R\$ 196,27	R\$ 442,00	R\$ 162,00

Figura 11 – Exemplo da tabela do exemplo 4 com o uso do atributo axis

Para mais detalhes veja o tutorial [Tabelas Acessíveis](#) (documento pdf - 154 KB) na seção do e-MAG no Portal do Programa de Governo Eletrônico.

RECOMENDAÇÃO 25 – GARANTIR A LEITURA E COMPREENSÃO DAS INFORMAÇÕES

Ver WCAG 2.0 Critério de Sucesso 3.1.5

O texto de um sítio deve ser de fácil leitura e compreensão, não exigindo do usuário um nível de instrução mais avançado do que o ensino fundamental completo. Quando o texto exigir uma capacidade de leitura mais avançada, deve ser disponibilizado informações suplementares que expliquem ou ilustrem conteúdo principal. Outra alternativa é versão simplificada do conteúdo em texto.

Existem algumas técnicas que auxiliam na melhora da inteligibilidade de textos, como, por exemplo:

- 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 ao invés 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.

Para mais informações sobre como escrever textos para web, acesse a Cartilha de Redação Web na Seção do e-PWG (Padrões Web em Governo Eletrônico) no Portal de Governo Eletrônico.

RECOMENDAÇÃO 26 – DISPONIBILIZAR UMA EXPLICAÇÃO PARA SIGLAS, ABREVIATURAS E PALAVRAS INCOMUNS

Ver WCAG 2.0 Critérios de Sucesso 3.1.3 e 3.1.4

Deve estar disponível uma explicação que identifique a forma completa ou o significado das abreviaturas e siglas. Para isso, pode ser utilizada a tag abbr.

Exemplo

```
<p>Bem-vindo à <abbr title="World Wide Web" lang="en">WWW</abbr>!</p>
```

Já as palavras que podem ser ambíguas, desconhecidas ou utilizadas de forma muito específica, deverão ser definidas através de um texto adjacente, uma lista de definições ou estarem contidas num glossário.

RECOMENDAÇÃO 27 – INFORMAR MUDANÇA DE IDIOMA NO CONTEÚDO

Ver WCAG 2.0 Critério de Sucesso 3.1.2

Se algum elemento de uma página possuir conteúdo em um idioma diferente do principal, este deverá estar identificado pelo atributo lang. Essa recomendação não se aplica para nomes próprios ou termos técnicos que sejam compreendidos no contexto.

Exemplo

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>
```

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>
```

2.4 Apresentação / Design

RECOMENDAÇÃO 28 – OFERECER CONTRASTE MÍNIMO ENTRE PLANO DE FUNDO E PRIMEIRO PLANO

Ver WCAG 2.0 Critério de Sucesso 1.4.3

As cores do plano de fundo e do primeiro plano deverão ser suficientemente contrastantes para que possam ser visualizadas, também, por pessoas com baixa visão, com cromodeficiências ou que utilizam monitores de vídeo monocromático.

Não deverão ser utilizadas imagens atrás do texto (background), pois acabam por dificultar a leitura e desviar a atenção do usuário.

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. A relação de contraste entre plano de fundo e primeiro plano de 3:1 é o nível mínimo de contraste recomendado pela ISO-9241-3. No entanto, levando-se em consideração a perda de percepção do contraste resultante da baixa acuidade visual, cromodeficiência ou perda de sensibilidade ao contraste devido

ao envelhecimento, é recomendada aqui uma maior relação de contraste, de, no mínimo, 4,5:1. Existem ferramentas gratuitas disponíveis na Web que verificam a relação de contraste entre as cores do plano de fundo e do primeiro plano, referenciadas no capítulo 4 deste documento e para uma tabela de cores no anexo 1.

RECOMENDAÇÃO 29 – NÃO UTILIZAR APENAS COR OU OUTRAS CARACTERÍSTICAS

SENSORIAIS PARA DIFERENCIAR ELEMENTOS

Ver WCAG 2.0 Critérios de Sucesso 1.3.3 e 1.4.1

A cor ou outras características sensoriais, como forma, tamanho, localização visual, orientação ou som não devem ser utilizadas como o único meio para transmitir informações, indicar uma ação, pedir uma resposta ao usuário ou distinguir um elemento visual.

Exemplo

HTML

```
<p>Existem três procedimentos para executar a tarefa:</p>
<ul>
  <li><a href="#">Procedimento A</a></li>
  <li><a href="#" class="recomendado">Procedimento B (Recomendado)</a></li>
  <li><a href="#">Procedimento C</a></li>
</ul>
```

CSS

```
a.recomendado{
  color: #FF0000;
}
```

Existem três procedimentos para executar a tarefa:

- [Procedimento A](#)
- [Procedimento B \(Recomendado\)](#)
- [Procedimento C](#)

Figura 12 – Exemplo correto de utilização de cores nos elementos

RECOMENDAÇÃO 30 – PERMITIR REDIMENSIONAMENTO DE TEXTO SEM PERDA DE FUNCIONALIDADE

Ver WCAG 2.0 Critério de Sucesso 1.4.4

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

Exemplo

Um sítio possui uma ferramenta que permite o redimensionamento do tamanho da fonte. Para isso existe um botão que aumenta a fonte de dois em dois pixels. Da mesma forma, existe um botão que diminui a fonte de dois em dois pixels e outro que retorna ao tamanho padrão da fonte. O sítio utiliza um leiaute flexível, isto é, à medida que a fonte aumenta ou diminui, o leiaute se ajusta automaticamente para que não ocorram “quebras”.

```
<title>Exemplo Básico de Leiaute Líquido</title>
<style type="text/css">
.coluna {
  border-left: 1px solid green;
  padding-left: 1%;
  float: left;
  width: 32%;
}
#rodape {
  border-top: 1px solid green;
  clear: both;
}
</style>
</head>
<body>

<h1>Exemplo de Redimensionamento</h1>
<form action="#">
  <fieldset>
    <legend>Redimensionamento do texto</legend>
    <input type="button" id="mais" value="Aumentar" />
    <input type="button" id="mais" value="Diminuir" />
    <input type="button" id="mais" value="Tamanho padrão" />
  </fieldset>
</form>

<h2>Este texto está dividido em três blocos...</h2>
<div class="coluna">
  <h3>Bloco 1</h3>
  <p>O objetivo desta técnica é apresentar o conteúdo sem que ocorra a introdução de uma barra horizontal.</p> </div>
<div class="coluna">
  <h3>Bloco 2</h3>
  <p>Este é um exemplo simples de leiaute de página que se adapta ao redimensionamento de texto.</p>
</div>
<div class="coluna">
```

```

<h3>Bloco 3</h3>
<p>Esta é uma técnica simples mas que pode ser implementada para leiautes mais complexos.</p>
</div>
<p id="rodape">Texto do Rodapé</p>
</body>
</html>

```

Exemplo de Redimensionamento

Redimensionamento do texto

Este texto está dividido em três blocos...

Bloco 1	Bloco 2	Bloco 3
O objetivo desta técnica é apresentar o conteúdo sem que ocorra a introdução de uma barra horizontal.	Este é um exemplo simples de leiaute de página que se adapta ao redimensionamento de texto.	Esta é uma técnica simples mas que pode ser implementada para leiautes mais complexos.

Texto do Rodapé

Figura 13 – Exemplo de página em seu tamanho padrão

Exemplo de Redimensionamento

Redimensionamento do texto

Este texto está dividido em três blocos...

Bloco 1	Bloco 2	Bloco 3
O objetivo desta técnica é apresentar o conteúdo sem que ocorra a introdução de uma barra horizontal.	Este é um exemplo simples de leiaute de página que se adapta ao redimensionamento de texto.	Esta é uma técnica simples mas que pode ser implementada para leiautes mais complexos.

Texto do Rodapé

Figura 14 – Exemplo de página redimensionada em 200% sem perda de funcionalidade

RECOMENDAÇÃO 31 – DIVIDIR AS ÁREAS DE INFORMAÇÃO

Ver WCAG 2.0 Critério de Sucesso 3.2.3 (Técnica G61)

Áreas de informação devem ser divididas em grupos fáceis de gerenciar. As divisões mais comuns são “topo”, “conteúdo”, “menu” e “rodapé”. Nas páginas internas deve-

se manter uma mesma divisão para que o usuário se familiarize mais rapidamente com a estrutura do sítio. É importante destacar, entretanto, que a página inicial pode ter uma divisão diferente das páginas internas, pois normalmente ela contém mais elementos. O exemplo a seguir mostra a divisão da página inicial de um sítio contendo os blocos "topo", "menu", "conteúdo" e "rodapé", além da barra de acessibilidade contendo os atalhos.

Exemplo

Topo

NOME DA INSTITUIÇÃO

Barra de Acessibilidade

- [Ir para conteúdo \[1\]](#)
- [Ir para menu principal \[2\]](#)
- [Ir para Busca \[3\]](#)

Menu

- Itens de menu
- ...

Conteúdo

Buscar

Pesquise aqui Pesquise aqui

BEM VINDO

Seja bem vindo ao nosso site.

Rodapé

Rua XXX

Figura 15 – Exemplo de divisão de blocos de conteúdo

```
<div id="topo">
  <a href="#inicioTopo" id="inicioTopo">Topo</a>
  <h1>NOME DA INSTITUIÇÃO</h1>

  <div id="barraAcessibilidade">
    <p>Barra de Acessibilidade</p>
    <ul>
      <li><a href="#inicioConteudo">Ir para conteúdo [1]</a></li>
      <li><a href="#inicioMenu">Ir para menu principal [2]</a></li>
      <li><a href="#busca">Ir para Busca [3]</a></li>
    </ul>
  </div>
</div>
<div id="menu">
  <a href="#inicioMenu" id="inicioMenu" accesskey="2">Menu</a>
  <ul>
    <li>Itens de menu</li>
    <li>...</li>
  </ul>
</div>
<div id="conteudo">
  <a href="#inicioConteudo" id="inicioConteudo" accesskey="1">Conteúdo</a>
  <form action="#" method="post">
    <fieldset>
      <legend>Buscar</legend>
      <label for="busca">Pesquise aqui</label>
```

```

<input type="text" id="busca" name="busca" accesskey="3" value="Pesquise aqui" />
<input type="submit" value="Buscar" class="buscar" name="buscar" />
</fieldset>
</form>
<h2>BEM VINDO</h2>
<p>Seja bem vindo ao nosso site.</p>
</div>
<div id="rodape">
<a href="#inicioRodape" id="inicioRodape">Rodapé</a>
<address> Rua XXX</address>
</div>

```

A divisão em blocos de conteúdo representa a base para a utilização dos atalhos, permitindo que o usuário tenha rápido acesso à área desejada. Para mais detalhes sobre a barra de acessibilidade e os atalhos, ver o *capítulo 3 – Padronização de acessibilidade nas páginas do Governo Federal*.

É importante que as diversas páginas de um sítio possuam um estilo de apresentação coerente e sistemático, mantendo-se um padrão de estrutura. Assim, elementos principais de navegação deverão ser mantidos na mesma posição em todas as páginas, com exceção da página inicial que, muitas vezes, apresenta uma estrutura diferenciada.

Exemplo

Um sítio possui um logotipo, um título, um formulário de pesquisa e uma barra de navegação. Esses elementos aparecem na mesma ordem relativa em cada página do sítio em que se repetem. Em uma das páginas, não há o formulário de pesquisa, mas o restante dos itens continua na mesma ordem.

RECOMENDAÇÃO 32 – POSSIBILITAR QUE O ELEMENTO COM FOCO SEJA VISUALMENTE EVIDENTE

Ver WCAG 2.0 Critério de Sucesso 2.4.7

A área que recebe o foco pelo teclado deve ser claramente marcada, devendo a área de seleção 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, mas não deverá ser removida.

Exemplo

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">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>
```

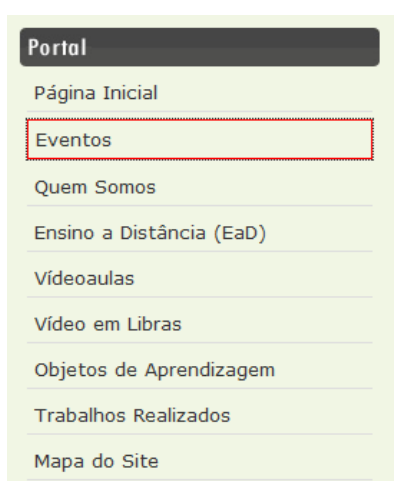


Figura 16 – Exemplo de foco visível em menu

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 passa-se 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`.

2.5 Multimídia

RECOMENDAÇÃO 33 – FORNECER ALTERNATIVA PARA VÍDEO

Ver WCAG 2.0 Critérios de Sucesso 1.2.1, 1.2.2, 1.2.6 e 1.2.8

Deve haver uma alternativa sonora ou textual para vídeos que não incluem faixas de áudio. Para vídeos que contêm áudio falado e no idioma natural da página, devem ser fornecidas legendas. Além de essencial para pessoas com deficiência visual, 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 ouvir um arquivo multimídia. Para tanto, podem ser fornecidos:

Situação 1

Um vídeo mostra como produzir uma tecnologia assistiva de baixo custo. Não há áudio, mas o vídeo inclui uma série de números para representar cada etapa do processo. Nesse caso, junto ao vídeo, deve ser disponibilizado um arquivo com a alternativa de texto que indica o conteúdo do vídeo e a descrição de cada uma das etapas.



Figura 17 – Vídeo contendo arquivo com alternativa em texto

Situação 2

Uma universidade oferece a opção de visualizar suas videoaulas com ou sem legendas.

Vídeoaulas:

- [A teoria da simplicidade \(com legenda\)](#)
- [A teoria da simplicidade \(sem legenda\)](#)
- [Dicas de HTML \(com legenda\)](#)
- [Dicas de HTML \(sem legenda\)](#)
- [Dicas de CSS \(com legenda\)](#)
- [Dicas de CSS \(sem legenda\)](#)

**Figura 18 – Vídeo com legenda**

Além de alternativa em texto e legenda, é desejável que os vídeos com áudio apresentem alternativa na Língua Brasileira de Sinais (Libras).

RECOMENDAÇÃO 34 – FORNECER ALTERNATIVA PARA ÁUDIO

Ver WCAG 2.0 Critérios de Sucesso 1.2.1, 1.2.2 e 1.2.6

Áudio gravado deve possuir uma transcrição descritiva. Além de essencial para pessoas com deficiência 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 ouvir um arquivo multimídia. Neste caso, também é desejável a alternativa em Libras.

Situação 1

Em um podcast o entrevistador faz perguntas a um especialista de saúde. Como essas informações são disponibilizadas ao usuário do sítio em um arquivo de áudio, deve ser fornecido um link para um arquivo com alternativa em texto, logo após o conteúdo em áudio.

Uma apresentação prévia do conteúdo dos dois tipos de arquivo e de sua duração também é desejável.

RECOMENDAÇÃO 35 – OFERECER AUDIODESCRIÇÃO PARA VÍDEO PRÉ-GRAVADO

Ver WCAG 2.0 Critério de Sucesso 1.2.3, 1.2.5 e 1.2.7

Vídeos que transmitem conteúdo visual que não está disponível na faixa de áudio devem possuir uma audiodescrição.

A audiodescrição consiste na descrição clara e objetiva de todas as informações apresentadas de forma visual e que não fazem parte dos diálogos. Essas descrições são apresentadas nos espaços entre os diálogos e nas pausas entre as informações sonoras.

Exemplo

Um vídeo de um malabarista se apresentando para um grupo de crianças inclui uma versão com audiodescrição. O narrador da audiodescrição descreve o número e o tipo de instrumentos que o malabarista utiliza, bem como as reações que as crianças têm durante a performance.

RECOMENDAÇÃO 36 – FORNECER CONTROLE DE ÁUDIO PARA SOM

Ver WCAG 2.0 Critério de Sucesso 1.4.2

Deve ser fornecido um mecanismo para parar, pausar, silenciar ou ajustar o volume de qualquer som que se reproduza na página.

RECOMENDAÇÃO 37 – FORNECER CONTROLE DE ANIMAÇÃO

Ver WCAG 2.0 Critério de Sucesso 2.2.2

Para qualquer animação que inicie automaticamente na página devem ser fornecidos mecanismos para que o usuário possa pausar, parar ou ocultar tal animação.

2.6 Formulários

RECOMENDAÇÃO 38 – FORNECER ALTERNATIVA EM TEXTO PARA OS BOTÕES DE IMAGEM DE FORMULÁRIOS

Ver WCAG 2.0 Critério de Sucesso 1.1.1

Ao serem utilizados botões do tipo imagem (input type="image"), que servem para o mesmo propósito do botão do tipo submit, deve ser fornecida uma descrição textual para o botão através do atributo alt, conforme o exemplo a seguir.

Exemplo 1



Figura 19 – Imagem de Botão enviar

Código:

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

Já para outros tipos de botões (reset e button), é preciso substituir o botão pela imagem que se deseja utilizar através do CSS. Nesse caso, para que o botão seja acessível, ele deve possuir um value descritivo, conforme o exemplo a seguir.

Exemplo 2



Figura 20 – Imagem de Botão Limpar

HTML

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

CSS

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

RECOMENDAÇÃO 39 – ASSOCIAR ETIQUETAS AOS SEUS CAMPOS

Ver WCAG 2.0 Critério de Sucesso 1.3.1 (Técnica H44)

As etiquetas de texto (label) devem estar associadas aos seus campos (input) correspondentes no formulário, através dos atributos for do label e id do input, os quais deverão ter o mesmo valor.

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>
```

RECOMENDAÇÃO 40 – ESTABELECEER UMA ORDEM LÓGICA DE NAVEGAÇÃO

Ver WCAG 2.0 Critério de Sucesso 2.4.3

Os elementos do formulário devem ser distribuídos corretamente através do código HTML, criando, assim, uma sequência lógica de navegação. Assim, os formulários devem primeiro ser codificados considerando a ordem lógica de navegação para depois serem organizados visualmente via CSS.

OBS: O atributo tabindex somente deverá ser utilizado quando existir real necessidade. É importante ressaltar que este atributo deverá ser utilizado com a semântica correta e deverá ser verificado manualmente se o fluxo fornecido pelo tabindex é realmente o desejado.

RECOMENDAÇÃO 41 – NÃO PROVOCAR AUTOMATICAMENTE ALTERAÇÃO NO CONTEXTO

Ver WCAG 2.0 Critério de Sucesso 3.2.2

Quando um elemento de formulário receber o foco, não deve ser iniciada uma mudança automática na página que confunda ou desorienta o usuário. Assim, as mudanças devem ocorrer através do acionamento de um botão.

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" />
```

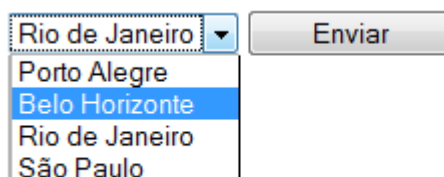
A imagem mostra um formulário web. À esquerda, há uma lista suspensa com o texto "Rio de Janeiro" no topo e uma seta para baixo. O menu está aberto, mostrando as opções "Porto Alegre", "Belo Horizonte", "Rio de Janeiro" e "São Paulo". À direita da lista, há um botão retangular com o texto "Enviar".

Figura 21 – Acionamento - forma correta

Exemplo Incorreto

```
<select name="cidade" id="cidade" onchange="location =  
this.options[this.selectedIndex].value;">  
  <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>
```

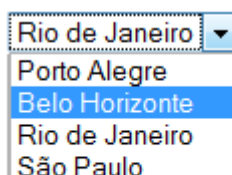
A imagem mostra apenas a parte da lista suspensa do formulário. O menu está aberto, mostrando as opções "Porto Alegre", "Belo Horizonte", "Rio de Janeiro" e "São Paulo".

Figura 22 – Acionamento - forma incorreta

RECOMENDAÇÃO 42 – FORNECER INSTRUÇÕES PARA ENTRADA DE DADOS

Ver WCAG 2.0 Critério de Sucesso 3.3.2

Para conteúdo que exigir entrada de dados por parte do usuário, devem ser fornecidas, quando necessário, instruções de preenchimento juntamente com as etiquetas (label). A utilização de caracteres pré-definidos em áreas de entrada de texto só deve ocorrer se:

- O texto for incluído após a entrada de dados pelo usuário (por exemplo, sugerir um novo nome de usuário caso o escolhido já exista);
- A semântica do documento justifique a inclusão de texto pré-definido (por exemplo, uma loja virtual que só vende para determinado país já vem com o campo país preenchido);
- Os caracteres tenham sido fornecidos previamente pelo usuário (por exemplo, refinamento de busca).

Exemplo

O seguinte exemplo indica que a data precisa ser inserida no formato dia (dd) –mês (mm) –ano (aaaa).

```
<label for="data">Data (dd-mm-aaaa)</label>  
<input type="text" id="data" name="data" />
```

RECOMENDAÇÃO 43 – IDENTIFICAR E DESCREVER ERROS DE ENTRADA DE DADOS

Ver WCAG 2.0 Critério de Sucesso 3.3.1

Quando um erro de entrada de dados for automaticamente detectado, o item que apresenta erro deve ser identificado e descrito ao usuário por texto.

Exemplo 1

Um usuário tenta submeter um formulário, mas esquece de preencher campos obrigatórios. Utilizando validação *client-side* (do lado do cliente), a omissão é detectada e um alerta aparece informando ao usuário que campos obrigatórios não foram preenchidos. Para a identificação destes campos, as etiquetas são modificadas tornando-se avisos e também o foco do teclado é remetido para o primeiro campo sem preenchimento.

Formulário de contato

É obrigatório o preenchimento dos campos com *.

Dados Pessoais

* Nome:
João
...

Dados Profissionais

Profissão:
Tec. Informática
...

Informações de Contato

* E-mail
[Campo vazio]
* Instituto:
IFFarroupilha

Enviar

[Aplicativo JavaScript]
O campo E-mail precisa ser preenchido
OK

Figura 23 – Alerta sobre o não preenchimento de campos obrigatórios

Formulário de contato

É obrigatório o preenchimento dos campos com *.

Dados Pessoais

* Nome:
João
...

Dados Profissionais

Profissão:
Tec. Informática
...

Informações de Contato

* E-mail: (O preenchimento é obrigatório!)
[Campo vazio]
* Instituto:
IFFarroupilha

Enviar

Figura 24 – Indicação do erro no formulário após a mensagem de alerta

Exemplo 2

Quando o usuário enviar o formulário e esquecer-se de preencher um campo obrigatório ou preencher um campo de maneira incorreta, o foco é remetido novamente ao formulário, mas que estará contendo apenas os campos com erro, para que o usuário não precise navegar por todos os outros campos novamente. Abaixo dos campos com erro que deverão ser novamente preenchidos, haverá o botão de envio e um botão para o usuário retornar ao formulário completo, caso ele deseje verificar todos os campos.

Exemplo 3

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.

RECOMENDAÇÃO 44 – AGRUPAR CAMPOS DE FORMULÁRIO

Ver WCAG 2.0 Critério de Sucesso 1.3.1 (Técnicas H71 e H85)

Deverão ser agrupados os controles de formulário relacionados de maneira lógica, utilizando-se o elemento fieldset, associando o elemento legend de forma significativa (o elemento fieldset é útil para agrupar, de forma lógica, elementos do formulário). Para cada fieldset, é possível fornecer uma legenda que explica claramente o propósito ou natureza dos agrupamentos.

Exemplo

```
<form method="post" action="...">
  <fieldset>
    <legend>Dados Pessoais</legend>
    <label for="nome">O seu Nome: </label>
    <input type="text" name="nome" id="nome" />
    ...
  </fieldset>
  <fieldset>
    <legend>Dados Profissionais</legend>
    <label for="profissao">Sua profissão: </label>
    <input type="text" id="profissao" name="profissao" />
    ...
  </fieldset>
  <fieldset>
    <legend>Informações de Contato</legend>
    <label for="email">E-mail: </label>
```

```

<input type="text" id="email" name="email" />
...
</fieldset>
</form>

```

Figura 25 – Imagem do Formulário

Para agrupar elementos option dentro de um elemento de lista select, deve ser utilizado o elemento optgroup.

Exemplo 2

```

<label for="instituto">Qual a sua instituição? </label>
<select id="instituto" name="instituto">
  <optgroup label="Rio Grande do Sul">
    <option value="ifrs">IFRS</option>
    <option value="ifsul">IFSUL</option>
    <option value="iffarroupilha">IFFarroupilha</option>
  </optgroup>
  <optgroup label="Santa Catarina">
    <option value="ifsc">IFSC</option>
    <option value="ifc">IFC</option>
  </optgroup>
  <optgroup label="Paraná">
    <option value="ifpr">IFPR</option>
  </optgroup>
</select>

```

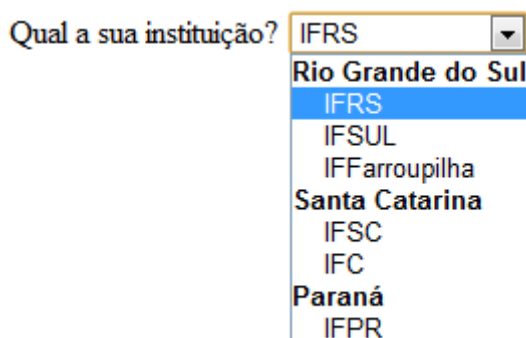



Figura 26 – Imagem da caixa de seleção

RECOMENDAÇÃO 45 – FORNECER CAPTCHA HUMANO

Ver WCAG 2.0 Critério de Sucesso 1.1.1 (Técnicas G143 e G144)

O CAPTCHA (teste interativo humano, completamente automatizado, para distinguir computadores de seres humanos), quando utilizado, deverá ser fornecido em forma de pergunta de interpretação. Tais perguntas poderão ser respondidas apenas por um ser humano. No entanto, é preciso garantir que a pergunta não seja de difícil resolução, permitindo que a mesma possa ser respondida por pessoas de variadas culturas e níveis de instrução. Para tal, podem ser utilizadas perguntas de senso comum, como por exemplo, “qual é a cor do céu?” ou “o fogo é quente ou frio?”. Também podem ser utilizados testes matemáticos. No entanto, é preciso tomar cuidado para que esses testes não sejam facilmente “quebrados” por determinados programas. Uma alternativa é solicitar que o usuário escreva o resultado do teste matemático por extenso, como “escreva por extenso quanto é $2 + 3$ ”, ou ainda “responda por extenso quanto é dois mais três”.

OBS: O CAPTCHA deverá ser utilizado apenas quando estritamente necessário.

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>
```

CAPTCHA

Escreva por extenso quanto é dois mais três.

Figura 27 – Exemplo de CAPTCHA

Para mais detalhes sobre o desenvolvimento de formulários acessíveis, veja o tutorial Formulários Acessíveis (arquivo pdf – 160 KB) na seção e-MAG no Portal do Programa de Governo Eletrônico.

3. Padrões de acessibilidade digital no Governo Federal

Esta seção pretende padronizar elementos de acessibilidade que devem ter características em comum em todas as páginas do governo federal, para fins de facilitar o acesso por pessoas com deficiência. Este conjunto de elementos busca garantir o desenvolvimento de sítios acessíveis a todas as pessoas.

Os elementos a serem padronizados, que devem estar presentes em todas as páginas do Governo Federal são:

1. Página com a descrição dos recursos de acessibilidade
2. Teclas de atalho
3. Barra de acessibilidade
4. Apresentação do mapa do sítio
5. Apresentação de formulário
6. Conteúdo alternativo para imagens
7. Apresentação de documentos

Ao final deste capítulo também serão apresentados elementos que não serão permitidos nas páginas do governo federal.

3.1 Página de descrição com os recursos de acessibilidade

Esta página apresenta os recursos de acessibilidade presentes no sítio, como as teclas de atalho disponíveis, as opções de redimensionamento de texto e alto contraste, detalhes sobre testes de acessibilidade realizados no sítio e outras informações pertinentes a respeito de sua acessibilidade. O link para a página contendo os recursos de acessibilidade deve ser disponibilizado na barra de acessibilidade, a qual será abordada no terceiro item desta seção.

Dicas de Acessibilidade

Tamanho do texto:

- aumentar letra
- diminuir letra
- tamanho normal

Alto contraste

- Preto
- Azul
- Verde
- Original

Teclas de acesso

Teclas de acesso são um recurso de navegação que permitem a você navegar neste web site com o seu teclado.

Teclas de acesso disponíveis

Este site usa uma configuração que se aproxima da maioria das recomendações internacionais sobre teclas de acesso. São elas:

1. Ir para Menu
2. Ir para Conteúdo
3. Ir para campo de busca

A combinação de teclas difere conforme o navegador utilizado. A seguir, encontram-se as indicações para ativar as teclas de atalho dos principais navegadores:

■ Para Windows e Linux

- Internet Explorer: [Alt] + [tecla de atalho]
- Mozilla Firefox e Netscape: [Alt] + [Shift] + [tecla de atalho]
- Opera: [Shift] + [Esc] + [tecla de atalho]

■ Para Macintosh

- Internet Explorer, Safari, Mozilla Firefox e Netscape: [Ctrl] + [tecla de atalho]
- Opera: [Shift] + [Esc] + [tecla de atalho]

Indicação de Acessibilidade

Nós nos responsabilizamos pelo uso de nosso conhecimento e entendimento das formas com que diferentes pessoas acessam a Internet, para desenvolver um web site que seja limpo e simples para que todos usem.

Validação

Nós estamos usando XHTML 1.1 e CSS que obedecem às especificações, como é proposto pela W3C porque nós acreditamos que acessibilidade e usabilidade devem ter um embasamento sólido. Se alguma coisa neste site não está validando corretamente, por favor entre em contato com a Administração do Site.

A validação da acessibilidade foi realizada por meio de ferramentas automáticas e de revisão direta através de programas leitores de tela. Os métodos automáticos são geralmente rápidos, mas não são capazes de identificar todas as vertentes da acessibilidade. Sendo assim, a avaliação humana ou manual ajuda a garantir a clareza da linguagem e a facilidade da navegação do usuário pelo site.

Figura 28 – Exemplo de página sobre acessibilidade do sítio

3.2 Atalhos de teclado

Deverão ser disponibilizados atalhos por teclado para pontos estratégicos da página, permitindo que o usuário possa ir diretamente a esses pontos. Os atalhos deverão funcionar através de números precedidos da tecla padrão de cada navegador (Alt no Internet Explorer, Shift + Alt no Firefox, Shift + Esc no Opera, etc.). 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;

As dicas dos atalhos deverão ser disponibilizadas na barra de acessibilidade, abordada a seguir e, também, na página sobre a acessibilidade do sítio, já comentada anteriormente. Para mais detalhes ver Recomendação 6.

3.3 Barra de acessibilidade

O sítio deverá conter uma barra de acessibilidade no topo de cada página contendo os seguintes itens:

- Aumentar fonte
- Diminuir fonte
- Fonte normal
- Alto contraste
- Atalhos (para Menu, Conteúdo e Busca)
- Acessibilidade (link para a página contendo os recursos de acessibilidade do sítio)

Exemplo de Barra de Acessibilidade

```
<div id="acessibilidade">
  <ul id="atalhos">
    <li><a href="#iniciodoconteudo">Ir para Conteúdo [1]</a></li>
    <li><a href="#iniciodomenu">Ir para Menu [2]</a></li>
    <li><a href="#busca">Ir para busca [3]</a></li>
  </ul>
  <ul id="botoes">
    <li><a href="#" id="btnAumentarFonte">aumentar letra</a></li>
    <li><a href="#" id="btnDiminuirFonte">diminuir letra</a></li>
    <li><a href="#" id="btnTamanhoFonte">tamanho normal</a></li>
    <li><a href="#" id="bt_contraste">alto contraste</a></li>
    <li><a href="acessibilidade.html">Página de acessibilidade </a></li>
  </ul>
</div>
```

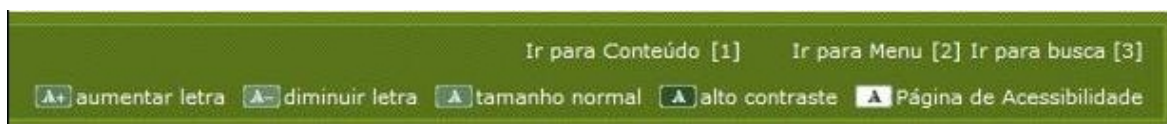


Figura 29 – Exemplo de página sobre acessibilidade do sítio

Quando uma das ferramentas aumentar fonte, diminuir fonte e fonte normal for utilizada, o bloco como um todo deve ser modificado, não apenas a fonte do texto.

A opção 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). As ferramentas utilizadas para verificar a relação de contraste entre as cores estão listadas no capítulo 4 – Recursos e Ferramentas para Acessibilidade.



Figura 30 – Exemplo de página com alto contraste

Para mais detalhes ver Recomendações 28 e 30.

3.4 Apresentação do mapa do sítio

Deverá ser fornecido um mapa do sítio para sítios que contenham páginas internas que não estão presentes no menu. O mapa do sítio deve ser disponibilizado em forma de lista, podendo conter quantos níveis forem necessários.



Figura 31 –Exemplo de mapa do sítio

3.5 Apresentação de formulário

Os formulários deverão estar de acordo com os seguintes itens:

- Sempre utilizar a tag form, mesmo que o formulário possua apenas um elemento, como é o caso de uma caixa para pesquisa.

- Disponibilizar os elementos do formulário no HTML na ordem correta de navegação, sem utilizar o tabindex.
- Associar as etiquetas (label) aos seus campos (input) correspondentes.
- Não deve ocorrer mudança no contexto quando um elemento receber o foco.
- Deve ser fornecido um botão de envio (submit) para enviar os dados. No entanto, é necessário fornecer uma maneira de o usuário poder verificar as informações antes que elas sejam enviadas.
- Os erros de entrada de dados devem ser identificados e descritos ao usuário.
- Para cada conjunto de informações, com dois ou mais elementos de entrada de dados, os mesmo deverão ser agrupados através do elemento fieldset/legend. Em um formulário de busca simples, por exemplo, não há a necessidade de utilizar o elemento fieldset/legend, pois apresenta apenas um campo de entrada de dados.

Exemplo de formulário de Fale Conosco

```
<form method="post" action="act/contato.php">
  <fieldset>
    <legend>Formulário de Contato</legend>
    <label for="nome">Nome:</label>
    <input type="text" name="nome" id="nome" />
    <label for="email">E-mail:</label>
    <input type="text" name="email" id="email" />
    <label for="assunto">Assunto:</label>
    <select name="assunto" id="assunto">

      <option value="0" selected="selected">Selecione o assunto desejado</option>
      <option value="informacoes">Informações</option>
      <option value="sugestoes">Sugestões</option>
      <option value="duvidas">Dúvidas</option>
    </select>

    <label for="msg">Mensagem:</label>
    <textarea name="msg" id="msg"></textarea>

    <input type="submit" name="enviar" id="enviar" value="Enviar" class="bnt" />

  </fieldset>
</form>
```


Você está em: [Página Inicial](#) > [Entre em contato](#)

[< VOLTAR](#) | [INFORMAR ERRO](#) | [IMPRIMIR](#) | [ENVIAR POR EMAIL](#)

Entre em Contato

Formulário de Contato
Nome:

E-mail:

Assunto:

Mensagem:

ENVIAR

Figura 32 – Formulário de contato

Após clicar no botão enviar, 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.

Você está em: [Página Inicial](#) > [Entre em contato](#) > [Página de Confirmação](#)

[< VOLTAR](#) | [INFORMAR ERRO](#) | [IMPRIMIR](#) | [ENVIAR POR EMAIL](#)

Página de Confirmação

Formulário de Contato
Nome: João de Almeida
E-mail: joao.almeida@provedor.com.br
Assunto: Informações
Mensagem: Teste

EDITAR ENVIAR

Figura 33 – Página de Confirmação

Para mais detalhes ver recomendações 38 a 45.

3.6 Conteúdo alternativo para imagens

Deverá ser fornecida uma alternativa textual, pelo atributo alt, para imagens, fotos, gráficos, banners, botões de imagem, áreas ativas de mapa de imagem, CAPTCHA, etc. Além do alt, para imagens mais complexas, que necessitem de uma descrição mais detalhada, deverá ser fornecida uma descrição longa no próprio contexto ou em um link (claramente identificado como descrição da imagem) logo após a imagem. Para mais detalhes ver recomendações 20 e 21.

3.7 Apresentação de documentos

Os documentos em texto deverão ser disponibilizados preferencialmente em HTML. Também podem ser disponibilizados arquivos para download no formato ODF, tomando-se os cuidados para que sejam acessíveis. Se houver necessidade de disponibilizar arquivos em PDF, deverá ser fornecida uma alternativa em HTML ou ODF. É necessário, também, informar a extensão e o tamanho do arquivo no próprio texto do link.

Para mais detalhes ver Recomendação 22.

3.8 Elementos que não devem ser utilizados

- Tabelas para fins de diagramação, conforme Recomendação 7;
- Atualizações automáticas periódicas, conforme Recomendação 11;
- Situações com intermitência de tela, conforme Recomendação 14;
- Elementos considerados depreciados pelo W3C, como frame, applet, blink, marquee, basefont, center, dir, align, font, isindex, menu, strike, u, etc.

4. Recursos e ferramentas para acessibilidade

Organizações e especificações

World Wide Web Consortium (W3C): <http://www.w3.org>

WCAG 2.0: <http://www.w3.org/TR/WCAG20>

WCAG 2.0 (português): <http://www.ilearn.com.br/TR/WCAG20>

Especificação HTML 5: <http://www.w3.org/TR/html5>

Especificação HTML 4.01: <http://www.w3.org/TR/html401>

Especificação XHTML 1.0: <http://www.w3.org/TR/xhtml1>

Especificação XHTML 1.1: <http://www.w3.org/TR/xhtml11>

Especificação CSS 1.0: <http://www.w3.org/TR/CSS1>

Especificação CSS 2.1: <http://www.w3.org/TR/CSS21>

Especificação CSS 3.0 (em construção): <http://www.w3.org/Style/CSS/current-work>

Codificação de caracteres: <http://www.w3.org/International/O-charset.pt-br.php>

Tutoriais W3C: <http://www.w3.org/2002/03/tutorials>

CSS Paged media: <http://www.w3.org/TR/REC-CSS2/page.html>

The Web Standards Project: <http://www.webstandards.org/>

Validadores de código

Validador (X)HTML: <http://validator.w3.org>

Validador CSS: <http://jigsaw.w3.org/css-validator>

Validador de links: <http://validator.w3.org/checklink>

Validadores automáticos de acessibilidade

ASES (WCAG e e-MAG):

<http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG/material-de-apoio>

Da Silva (WAI e e-MAG): <http://www.acessobrasil.org.br>

Cynthia Says (WAI): <http://www.cynthiasays.com>

Hera (WAI): <http://www.sidar.org/hera/>

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

TAW (WAI): <http://www.tawdis.net>

eExaminator (WCAG 1.0): <http://www.acesso.unic.pt/examinator.php>

Functional Accessibility Evaluator 1.1: <http://fae.cita.uiuc.edu/>

Checklists para validação humana

Checklist Manual de Acessibilidade - Desenvolvedores (documento pdf - 552 KB)

<http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG/material-de-apoio>

Checklist Manual de Acessibilidade – Pessoas com deficiência visual (documento pdf -

333 KB)

<http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG/material-de-apoio>

Ferramentas para análise de relação de contraste

Luminosity Colour Contrast Ratio Analyser (online):

<http://juicystudio.com/services/luminositycontrastratio.php>

Color Contrast Analyser (online): <http://www.colorsontheweb.com/colorcontrast.asp>

Check mu colours (online): <http://www.checkmycolours.com/>

Contrast Analyser 2.2: <http://www.paciellogroup.com/resources/contrast-analyser.html>

Simulador de Navegador

Browsershots: <http://browsershots.org>

Leitores de Tela

Jaws for Windows - Leitor de tela americano produzido pela Freedom Scientific:

<http://www.freedomscientific.com>

NVDA - Leitor de tela gratuito e de código aberto para Windows: <http://www.nvda-project.org>

Orca - Leitor de tela gratuito e de código aberto para Linux:

<http://live.gnome.org/Orca>

Virtual Vision - Leitor de tela nacional fabricado pela Micropower:

<http://www.micropower.com.br>

VoiceOver - Leitor de tela para MAC OS:

<http://www.apple.com/br/accessibility/voiceover> Windows Eyes - Leitor de tela canadense fabricado pela GW Micro: <http://www.gwmicro.com/Window-Eyes>

DOSVOX: Interface especializada desenvolvida pela UFRJ: <http://intervox.nce.ufrj.br>

Descrição dos Leitores de Tela (documento pdf - 2.71 MB)

<http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG/material-de-apoio>

Avaliação dos Pontos de Fragilidades em Leitores de Tela (documento pdf - 1.47 MB)

<http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG/material-de-apoio>

Navegadores Textuais

Lynx: <http://lynx.browser.org>

Lynx Viewer (simulador): <http://www.delorie.com/web/lynxview.html>

Simulador para Cegueira Cromática (Daltonismo)

Vischeck: <http://www.vischeck.com/>

Tutoriais

O uso correto do texto alternativo (documento pdf - 239 KB)

<http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG/material-de-apoio>

Tabelas Acessíveis (documento pdf - 153 KB)

<http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG/material-de-apoio>

Formulários Acessíveis (documento pdf - 159 KB)

<http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG/material-de-apoio>

5. Leitura complementar

Padrões Brasil e-Gov

<http://www.governoeletronico.gov.br/acoes-e-projetos/padroes-brasil-e-gov>

WCAG 2.0

<http://www.ilearn.com.br/TR/WCAG20>

Técnicas para as WCAG 2.0

<http://www.acesso.unic.pt/w3/TR/WCAG20-TECHS/>

CSS e Web Standards

<http://maujor.com/>

RENAPI – Projeto de Acessibilidade Virtual

<http://bento.ifrs.edu.br/acessibilidade/>

Acessibilidade Web: (X)HTML, CSS, Scripts e Usabilidade para Todos

<http://www.acessibilidadelegal.com/>

Acesso Digital

<http://www.acessodigital.net>

Web Accessibility in Mind

<http://webaim.org>

W3C Brasil

<http://w3c.br>

CLF Standards (Canadá)

<http://www.tbs-sct.gc.ca/clf2-nsi2/index-eng.asp>

Section 508 (Estados Unidos)

<http://www.section508.gov/>

NDA (Irlanda)

<http://www.nda.ie/cntmgmtnew.nsf>

6. Glossário

Acessibilidade: Significa permitir o acesso por todos, independente do tipo de usuário, situação ou ferramenta.

Acessibilidade à Web: Criar ou tornar as ferramentas e páginas Web acessíveis a um maior número de usuários, inclusive pessoas com deficiência.

Agente de Usuário: Qualquer software que recupera e apresenta conteúdo Web para seus usuários, possibilitando que o usuário interaja com o conteúdo Web.

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

CSS (Cascading Style Sheets – Folhas de estilo em cascata): Linguagem que descreve a apresentação, visual ou sonora, de um documento.

Leitor de tela: Software que fornece informações através de síntese de voz sobre os elementos exibidos na tela do computador. Esses softwares interagem com o sistema operacional, capturando as informações apresentadas na forma de texto e transformando-as em resposta falada através de um sintetizador de voz. Para navegar utilizando um leitor de tela, o usuário faz uso de comandos pelo teclado. O leitor de tela também pode transformar o conteúdo em informação tátil, exibida dinamicamente em Braille por um hardware chamado de linha ou display Braille, servindo, em especial, a usuários com surdocegueira.

Navegador: Programa que permite ao usuário consultar e interagir com o material publicado na Web.

Navegador textual: Tipo de navegador baseado em texto, diferente dos navegadores com interface gráfica, onde as imagens são carregadas. O navegador textual pode ser utilizado com leitores de tela por pessoas com deficiência visual e, também, por pessoas que acessam a Internet com conexão lenta.

Padrões Web (Web Standards): Padrões Web são regras de tecnologia definidas pelo W3C e outros órgãos da área utilizados para criar e interpretar o conteúdo Web.

Podcast: Arquivo ou uma série de arquivos de áudio digital, geralmente em formato MP3 ou AAC (este pode conter imagens estáticas e links), publicada através de

podcasting (forma de publicação de arquivos de mídia digital pela Internet, através de um feed RSS).

Tag: Código usado para marcar o início e o fim de um elemento (X)HTML.

Tecnologia assistiva: refere-se ao conjunto de artefatos disponibilizados às pessoas com deficiência, que contribui para prover-lhes uma vida mais independente, com mais qualidade e possibilidades de inclusão social. O propósito das tecnologias assistivas reside em ampliar a comunicação, a mobilidade, o controle do ambiente, as possibilidades de aprendizado, trabalho e integração na vida familiar, com os amigos e na sociedade em geral.

W3C (World Wide Web Consortium): Consórcio de âmbito internacional com a finalidade de desenvolver especificações, guias e ferramentas para Web.

WAI: (Accessibility Initiative - Iniciativa de Acessibilidade na Web): Tem como objetivo desenvolver protocolos que melhorem a interoperabilidade e promovam a evolução da WWW.

WCAG (Web Content Accessibility Guidelines: Recomendações de Acessibilidade para Conteúdo Web): Abrange diversas recomendações com a finalidade de tornar o conteúdo Web mais acessível.

Webcast: Transmissão de áudio e vídeo por meio da Internet ou redes corporativas para um grupo de pessoas, simultaneamente.

XHTML (Extensible HyperText Markup Language): É a linguagem HTML reformulada para seguir as regras do XML.

XML (Extensible Markup Language): Uma linguagem de marcação semelhante à HTML, mas que permite a descrição de dados definidos por elementos de programação.

7. Anexo 1 – Tabela de Contraste de Cores

As tabelas a seguir mostram as cores que, em fundo preto ou branco, apresentam uma relação de contraste de no mínimo 4,5:1.

Cores contrastantes com preto					
#090	#093	#096	#099	#09C	#09F
#0C0	#0C3	#0C6	#0C9	#0CC	#0CF
#0F0	#0F3	#0F6	#0F9	#0FC	#0FF
#390	#393	#396	#399	#39C	#39F
#3C0	#3C3	#3C6	#3C9	#3CC	#3CF
#3F0	#3F3	#3F6	#3F9	#3FC	#3FF
					#66F
#690	#693	#696	#699	#69C	#69F
#6C0	#6C3	#6C6	#6C9	#6CC	#6CF
#6F0	#6F3	#6F6	#6F9	#6FC	#6FF
			#969	#96C	#96F
#990	#993	#996	#999	#99C	#99F
#9C0	#9C3	#9C6	#9C9	#9CC	#9CF
#9F0	#9F3	#9F6	#9F9	#9FC	#9FF
					#C0F
			#C39	#C3C	#C3F
#C60	#C63	#C66	#C69	#C6C	#C6F
#C90	#C93	#C96	#C99	#C9C	#C9F
#CC0	#CC3	#CC6	#CC9	#CCC	#CCF
#CF0	#CF3	#CF6	#CF9	#CFC	#CFF
#F00	#F03	#F06	#F09	#F0C	#F0F
#F30	#F33	#F36	#F39	#F3C	#F3F
#F60	#F63	#F66	#F69	#F6C	#F6F
#F90	#F93	#F96	#F99	#F9C	#F9F
#FC0	#FC3	#FC6	#FC9	#FCC	#FCF
#FF0	#FF3	#FF6	#FF9	#FFC	#FFF

Tabela 1 – Cores contrastantes com a cor preta

Cores contrastantes com branco

#000	#003	#006	#009	#00C	#00F
#030	#033	#036	#039	#03C	#03F
#060	#063	#066	#069	#06C	#06F
#300	#303	#306	#309	#30C	#30F
#330	#333	#336	#339	#33C	#33F
#360	#363	#366	#369	#36C	#36F
#600	#603	#606	#609	#60C	#60F
#630	#633	#636	#639	#63C	#63F
#660	#663	#666	#669	#66C	#66F
#900	#903	#906	#909	#90C	#90F
#930	#933	#936	#939	#93C	#93F
#960	#963	#966			
#C00	#C03	#C06	#C09	#C0C	
#C30	#C33	#C36	#C39		

Tabela 2 – Cores contrastantes com a cor branca