

TECNOLOGIAS WEB

ACESSIBILIDADE

Olá!

Nesta aula, iremos aprender como construir e utilizar os conceitos e tags XHTML com foco na acessibilidade, garantindo uma universalização do acesso às páginas que se venha a construir. Veremos, ainda, como aperfeiçoar nosso código para uma maior flexibilidade dos conteúdos on-line utilizando esses recursos.

Ao final desta aula, você será capaz de:

- Apresentar os principais problemas de Aula 08: Acessibilidade na Internet
- Apresentar as principais deficiências e as tecnologias de apoio
- Apresentar as diretrizes do W3C para a Aula 08: Acessibilidade
- Compreender as principais de soluções em HTML para transformar seu site
- Listar os principais aplicativos utilizados neste segmento

1 O que é acessibilidade

Falar sobre a acessibilidade, principalmente na Internet, é falar em romper barreiras para o acesso às informações.

Todos têm o direito de ver, de ouvir e de sentir o que está acontecendo ao redor de si mesmo.

A preocupação em garantir o total acesso às informações deveria ser meta tanto nas empresas quanto em todas as esferas governamentais.

Entretanto, o que se observa é que existem portais com ações de cidadania que nem ao menos são acessíveis. O cliente mais engajado com esses princípios observa que as ações sociais mantidas pela empresa não estão preocupadas com o básico, ou seja, a acessibilidade.

No mundo, isso está mudando, pois cada empresa está sendo “avaliada pelo mercado” pelo conjunto social de sua obra e não somente por alguma ação com retorno social.

De acordo com a Lei 5296 de dezembro/2004

“Condição para utilização, com segurança e autonomia, total ou assistida, dos espaços, mobiliários e equipamentos urbanos, das edificações, dos serviços de transporte e dos dispositivos, **sistemas e meios de comunicação e informação, por pessoa portadora de deficiência ou com mobilidade reduzida.**”

Saiba mais



Para mais informações, leia, agora, os seguintes textos:

- **Estatísticas:** http://estaciodocente.webaula.com.br/cursos/GRA270/doc/aula08_estatisticas.pdf
- **Tipos de Deficiências & Soluções Tecnológicas:** http://estaciodocente.webaula.com.br/cursos/GRA270/doc/aula08_tiposdedeficiencias.pdf
- **Legislação:** http://estaciodocente.webaula.com.br/cursos/GRA270/doc/aula08_legislacao.pdf

2 Acessibilidade e a Internet

Ao longo dos anos, tanto o W3C (Consórcio Mundial de Internet) quanto o WAI (Projeto de Acessibilidade na Internet) não só têm apontado os problemas quanto a esse tema mas também têm feito sugestões de modificações nos portais e nos conteúdos para a Internet a fim de garantir que problemas como os descritos abaixo não mais existam.

Problemas apontados pelo W3C e WAI:

2.1 Problema 1

Na Internet, é real a dificuldade que o deficiente visual tem em ler os textos devido às características dos mesmos. Existem os textos criados a partir de imagens, existem os textos cuja formatação impede o acesso de pessoas com diversos tipos de problemas visuais (baixa visão, cegueira ou problemas com contrastes) como, por exemplo, a sintaxe matemática e, também, existem os textos gerados dinamicamente por recursos que dependem de sistemas instalados na máquina do usuário (Javascript), impedindo o acesso, a qualquer tempo, por quem quer que seja.

Além disso, existem os textos sem representação fonética, o que causa dificuldade de entendimento dos mesmos para aqueles que nunca ouviram a nossa língua, como é o caso dos surdos.

Exemplo 1:

Suponha que x seja maior ou igual a 0.

Escrevendo com a simbologia matemática usual, teremos: $x \geq 0$.

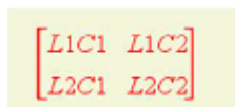
Se pensássemos na acessibilidade, poderíamos reescrever de maneira semelhante, porém universal: $x \geq 0$.

Dessa forma, todo navegador poderia apresentar esse conteúdo sem a necessidade de se instalar fontes e outros recursos.

Exemplo 2:

Outro exemplo mais complexo, porém adaptável, é o uso de matrizes que se apresentariam como na figura.

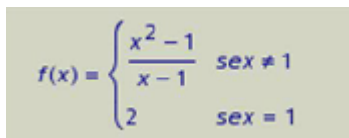
Poderíamos, na forma acessível, ter o mesmo conteúdo utilizando a seguinte sintaxe: [(L1C1, L1C2),(L2C1, L2C2)]


$$\begin{bmatrix} L1C1 & L1C2 \\ L2C1 & L2C2 \end{bmatrix}$$

Exemplo 3:

Utilizando a sintaxe do MATHML, definida pelo W3C, poderíamos, ao invés da função descrita de forma gráfica como na figura, ter, na forma acessível, a mesma função, só que utilizando o formato textual:

$f(x) = \{((x^2 - 1)/(x - 1), \text{ se } x \neq 1), (2, \text{ se } x = 1)\}$


$$f(x) = \begin{cases} \frac{x^2 - 1}{x - 1} & \text{se } x \neq 1 \\ 2 & \text{se } x = 1 \end{cases}$$

2.2 Problema 2

Existem dificuldades em se interpretar as informações seja pela barreira da língua, seja pela forma de exposição ou pela busca de informações cujas indicações, muitas vezes, não são dadas ou não estão claras por se acreditar ser óbvio o que se quer dizer.

Nesse caso, é óbvio para quem fez ou para todos os internautas? É fundamental que os especialistas no assunto não validem o que foi construído a fim de garantir o perfeito entendimento e acesso de todos.

2.3 Problema 3

Existem pessoas que, por razões diversas, não podem utilizar o mouse. Muitos portais somente permitem interação do usuário via mouse, solicitando-lhe “clique para pular a introdução”. Se ele não pode utilizar o mouse, ele nunca poderá interagir.

Outro problema, ainda, são os teclados virtuais que, embora disponibilizados por medida de segurança, não fornecem alternativas àqueles que não podem utilizar o mouse. Nesse caso, estamos falando de impossibilidades cuja complexidade varia como, por exemplo, um simples cabo de mouse partido, pessoas portadoras de LER (Lesão por Esforço Repetitivo) e pessoas com paralisia.

2.4 Problema 4

Existem portais que foram projetados para garantir a atenção dos usuários e, no caso de tarefas em que isto não é possível, não apresentam outras alternativas.

Imagine-se sendo capaz de ler as informações de um portal enquanto dirige um carro. Nada feito. Mas, se o portal disponibilizasse alguma alternativa que tornasse isso possível, ou se o dispositivo de conexão web solucionasse esse problema, mesmo dirigindo, poder-se-ia buscar, na internet, informações através de comandos de voz, e a resposta de tal busca chegaria, também, através de um conteúdo falado. Para isso, tanto os portais quanto os dispositivos móveis deveriam prever esse tipo de solução.

2.5 Problema 5

Necessidade de frequente atualização de recursos na máquina do usuário.

Sabemos que as inovações tecnológicas são importantes, mas, em muitos casos, o mais importante é o conteúdo, a informação dentro do portal e não seus recursos funcionais que obrigam o usuário a atualizar os próprios sistemas.

Imagine que uma pessoa, navegando pela Internet, ficou empolgada para comprar um carro, mas utiliza um sistema que não possui a última versão de um determinado software ou ainda não possui aquele software proprietário para acessar o conteúdo do portal da empresa para escolher o modelo do carro. Isso acontece a toda hora. Muitas vezes, o cliente altera a escolha que fez pela falta de preocupação da empresa em garantir o acesso de qualquer pessoa ao conteúdo do portal. Imagine o que pode acontecer ao desenvolvedor se o cliente deixar de comprar um carro de trezentos mil reais porque ele pressupõe que o cliente em potencial para esse tipo de automóvel deva ter um determinado sistema instalado em sua máquina.

No dia em que as empresas souberem que isso acontece, deve-se lamentar pelos desenvolvedores e pelos web designers responsáveis pela criação desses portais.

Esses são, apenas, alguns dos problemas que ocorrem, de maneira frequente e recorrente, nos portais. Todos aqueles que, de alguma forma, lidam com tecnologias devem observar esses problemas e fazer as devidas adaptações para que, sem exceção, qualquer pessoa com qualquer tecnologia possa aproveitar todas as informações contidas nos portais da Internet.

Para isso, é necessário que o desenvolvedor aprenda quais são os recursos que estão a disposição dos deficientes.

3 Sistemas auxiliares

Existem sistemas capazes de auxiliar o usuário tanto na sua navegação pela Internet quanto na interatividade com dispositivos.

São eles:

- Computadores

Hoje, mais do que nunca, a utilização do computador como recurso de auxílio na construção do conhecimento garante ao usuário contato com o mundo.

O computador não exige locomoção para ser usado e, por isso, tornou-se a extensão dos olhos, dos ouvidos, das pernas e dos braços dos usuários com deficiência.

Existem políticas públicas que facilitam a aquisição de computadores e de sistemas para esse tipo de usuário.

- Sintetizadores de voz

Esse recurso é muito empregado em consorciação de computadores, pois permite que tanto cegos quanto deficientes motores possam interagir com as pessoas através deles, rompendo, assim, muitas barreiras.

Atualmente, existem sintetizadores de voz bastante eficientes que apresentam entonações diferenciadas e funcionam como multilíngues garantindo não só a locução do conteúdo apresentado como também o reconhecimento de comandos através da voz humana.

- Sistema Braille

Esse sistema de leitura pelo tato, criado por Louis Braille, em 1806, permite ao deficiente visual ler e escrever utilizando um conceito bastante simples de cela Braille, onde aparecem 2 colunas com 3 pontos cada uma.

A combinação desses pontos forma letras, números e símbolos e permite ao deficiente se apropriar das informações escritas e impressas quando se faz uso desse tipo de recurso.

Existe uma folha de estilo especial para este propósito que, tal qual páginas com layouts de impressão, indica ao dispositivo Braille que aquele conteúdo é adaptado para ele sem a necessidade de impressão do mesmo.

Outra informação importante é que nem todos os deficientes visuais dominam o Braille pelas dificuldades do aprendizado, principalmente para aqueles que perderam recentemente a visão.

- Impressora Braille

O recurso da impressão em Braille proporciona maior conforto ao usuário, pois não há necessidade de dispositivos especiais para leitura. Contudo, a conversão do conteúdo para o Braille apresenta algumas dificuldades como, por exemplo, as imagens, os gráficos e as expressões matemáticas não adaptadas. Além disso, há, também, o alto custo das impressoras Braille.

Hoje, uma impressora jato de tinta pode custar algo em torno de duzentos reais enquanto uma impressora Braille pode custar mais de quinze mil reais.

Outro dado importante é a conversão de uma lauda impressa para Braille. A equivalência de cada lauda impressa pode variar entre 2 e 4 folhas em Braille. Imagine o peso de um livro neste sistema.

- Ampliação do conteúdo

Existem formas de se ampliar o conteúdo, quer sejam imagens, quer sejam textos, para aqueles que possuem baixa visão.

O problema de baixa visão requer que a impressão ampliada seja feita de acordo com a característica do problema de cada usuário, isto é, existem diversos graus de comprometimento da visão que levam, por exemplo, à necessidade de textos com letras de tamanho 72. Isso mesmo, o número não está errado, 72 e equivale a uma letra da grossura de um polegar.

- Libras

Os deficientes auditivos que não possuem implante coclear e não sabem ler lábios necessitam de um sistema que seja capaz de traduzir para a linguagem brasileira de sinais (LIBRAS) o que está sendo representado em áudio, em vídeo e, até, em texto.

Neste ponto, você pode estranhar o porquê da tradução do texto para a LIBRAS se o surdo enxerga. Ele pode ler, sim, mas interpretar a fonética das palavras dando sentido ao texto é uma outra história. Por isso, alguns programas televisivos dispõem de intérpretes de Libras que aparecem no canto da tela ao invés de, somente, se colocar o texto com recursos de legenda do tipo **close caption**.

4 Produtos e soluções

- **Sistema DOSVOX** No Brasil, há produtos como o DOSVOX, criado pela equipe do Prof. Antônio Borges do Núcleo de Computação Eletrônica da UFRJ e utilizado, no mundo, por mais de 30 mil usuários, que possui uma suíte de aplicativos bastante diversificados e frequentemente atualizados que auxiliam o deficiente visual não só a navegar pela Internet como também no próprio uso que faz do computador no dia-a-dia. A limitação do DOSVOX é o reconhecimento automático de Javascript e, com isso, o uso de tecnologias baseadas em sistemas clientes como o AJAX, por exemplo, fica prejudicado. Esse tipo de limitação é bastante comum na maioria dos navegadores para dispositivos móveis. Existem alternativas técnicas bastante razoáveis que permitem aos portais não serem clientes-dependentes.
- **Sistema Motrix** É um sistema que permite aos portadores de qualquer tipo de deficiência motora, grave ou não, o acesso a um computador, possibilitando-lhes a leitura escrita e a interação com outras pessoas através da Internet.
- **Braille Fácil** Sistema criado pela equipe do Prof. Antônio Borges do Núcleo de Computação Eletrônica da UFRJ para auxiliar na conversão de texto para Braille. O Braille Fácil é gratuito e possui, na 3ª geração, recursos de imagens. intervox.nce.ufrj.br/brfacil/
- **Virtual Vision** Desenvolvido no Brasil, este produto é comercializado sob licença especial, pois visa auxiliar o deficiente visual na navegação pela Internet e na interatividade com os aplicativos do computador pessoal. Clientes deficientes visuais de alguns bancos recebem-no gratuitamente.
- **Monet** Sistema desenvolvido pela Acessibilidade Brasil, integrável ao Braille Fácil, capaz de converter imagens em pontos através da utilização de impressoras Braille.
- **Dicionário de Libras** Sistema desenvolvido pela Acessibilidade Brasil que traz o dicionário de Libras para a Internet. Este sistema não interpreta textos. Ele é apenas um dicionário.
- **Rybená** Sistema interpretador de LIBRAS. É um produto comercial desenvolvido pela equipe do DFJUG – Desenvolvedores Java do Distrito Federal. Este Applet em Java pode ser acoplado ao portal e garantir aos surdos uma navegação de melhor qualidade.
- **Dirce** É um programa gerador e tocador de audiolivros baseado na tecnologia padrão DAISY para livros em formato digital com narração em áudio.
- **JAWS** Leitor de tela comercial. Muito utilizado pelos deficientes visuais. Interpreta javascript. freedomscientific.com/products/software/jaws/
- **Windows Eve** Leitor de tela comercial. Muito utilizado pelos deficientes visuais. Interpreta javascript.
- **NVDA** Leitor de tela gratuito. Muito utilizado pelos deficientes visuais. Interpreta javascript. nvaccess.org/
- **Complex Table Toolbar** Conversor de Tabelas. A equipe do Vision Australia criou uma barra de ferramenta para o Mozilla Firefox que valida a acessibilidade nas tabelas e propõe alterações, no próprio código, para adaptá-las.
- **Opera** Navegador com opções muito completas para acessibilidade auxiliando tanto desenvolvedores quanto usuários finais na Web.

5 Diretrizes de acessibilidade para o conteúdo da Web (WCAG) 2.0

As diretrizes de acessibilidade propostas pelo comitê do W3C vêm, desde 1999, abordando diversos tipos de problemas e de soluções para garantir acesso universal a todo e qualquer internauta que utilize dispositivos, móveis ou não, que obedeçam a essas diretrizes.

1999

Em 1999, surge a versão 1.0 chamada de WCAG 1.0.

2008

Com o advento da versão 2.0 em 2008, os critérios de sucesso são apresentados como declarações testáveis que não são de tecnologia específica. São orientações de como interpretar os critérios de sucesso e sobre como satisfazê-los em tecnologias específicas.

- Mas o que é WCAG?

A sigla WCAG representa Web Content Accessibility Guidelines. Em uma tradução livre, teremos: Guia de Acessibilidade aos Conteúdos na Web.

- A quem se destina o WGAG?

O WGAG se destina, principalmente, ao público dos:

- Desenvolvedores de conteúdo Web (autores de páginas, designers de sites, etc.).
- Desenvolvedores de ferramentas de autoria Web.
- Desenvolvedores de ferramentas de avaliação de acessibilidade para a Web.
- Outros que desejam ou que precisam de uma norma técnica de acessibilidade na Web.

6 Os princípios do WGAG 2.0

WCAG 2.0 tem 12 diretrizes que são organizadas em 4 princípios:

- 1. Perceptível
- 2. Operável
- 3. Compreensível
- 4. Robusto

Abordaremos, no curso, os 2 primeiros princípios indicando o próprio WCAG 2.0 como referência para a complementação do aprendizado.

Para cada orientação, existem critérios de sucesso testáveis que são em três níveis: A, AA e AAA.

Quanto maior o número de letras A, maior o grau de acessibilidade apresentado, isto é, mais princípios foram atendidos.

7 Princípio 1 – Perceptível

7.1 Fornecer alternativas de texto para conteúdo não texto

- **Alternativas textuais**

O desenvolvedor deve fornecer alternativas textuais a qualquer conteúdo não-textual em sua página, permitindo, assim, que o mesmo possa ser acessível ou alterado em outras formas mais adequadas à necessidade da pessoa.

Exemplo: Impressão em Braille ou ampliada.

- **Conteúdo não textual**

Para conteúdos não-textuais, a solução é a apresentação de uma alternativa em texto que substitua ou indique o que se espera no caso apresentado, exceto para as situações indicadas abaixo.

1. Controles de Entrada: Se o conteúdo não-textual for um controle ou aceitar a entrada de dados, então faz-se necessário indicá-lo por um nome que descreva, de forma suficiente, sua finalidade.

2. Recursos Multimídia Baseados em Tempo

Caso se utilize algum recurso dessa espécie, faz-se necessário que as alternativas textuais forneçam uma identificação descritiva desse conteúdo.

3. Teste

Caso o conteúdo não-textual seja um teste ou um exercício, então deve-se indicar as alternativas em texto, fornecendo, no mínimo, uma identificação descritiva do conteúdo.

4. Sensorial

Caso queira criar uma experiência sensorial específica através do conteúdo não-textual, faz-se necessário, então, que as alternativas em texto forneçam, ao menos, uma identificação descritiva do conteúdo.

5. CAPTCHA

É muito utilizado para se certificar que o acesso ao recurso é feito por uma pessoa e não por um computador. Com isso, são fornecidas as alternativas textuais que identificam e descrevem a finalidade do conteúdo tornando disponíveis, assim, as formas alternativas ao CAPTCHA que utiliza saídas diferentes tais como sons.

6. Decoração, Formatação, Invisível

Caso queira utilizar-se de conteúdos não-textuais apenas para formatação visual ou para que não sejam apresentados nos navegadores, é necessário, então, fazê-lo de forma que possa ser ignorado pelo dispositivo em uso.

7.2 Fornecer legendas e alternativas de conteúdo de áudio e vídeo

-

Multimídia Baseada em Tempo

Cabe ao desenvolvedor fornecer alternativas para os recursos multimídia baseados em tempo.

- Apenas Áudio e Apenas Vídeo (Pré-gravados)

Apenas Áudio Pré-gravado: O WCAG 2.0 indica que deve ser fornecida uma solução a esse tipo de recurso para apresentar informações equivalentes ao conteúdo em áudio pré-gravado.

- Apenas Vídeo Pré-gravado

Deve ser fornecida uma solução em áudio que apresente informações equivalentes para o conteúdo em vídeo pré-gravado.

7.3 Fazer um conteúdo adaptável e disponibilizá-lo para tecnologias assistíveis

Adaptável

O WGAG 2.0 recomenda a criação de conteúdos que possam ser apresentados de diferentes maneiras para que não se percam as informações ou a estrutura.

7.4 Uso de contraste suficiente para tornar as coisas fáceis de se ver e de se ouvir

- **Discernível** Faz-se necessária a criação de alternativas, como a audição e a visualização de conteúdos, para a separação do primeiro plano e do plano de fundo no navegador do usuário.
- **Utilização de Cores** Recomenda-se que a cor não seja utilizada como o único meio visual de transmitir informações, identificar visualmente um elemento ou solicitar uma resposta ao usuário.

8 Princípio 2 - Operável

8.1 Permitir que todas as funcionalidades sejam acessíveis via teclado

Acessível por Teclado

É fundamental, para se ter acesso às funcionalidades do conteúdo de um portal, que se possa fazê-lo não só através do uso do mouse mas também através do uso do teclado. Toda a funcionalidade através desses tipos de dispositivos necessita de temporizações específicas para digitação de informações.

Dependendo do elemento em HTML, o texto já se encontra associado através de um atributo obrigatório. No caso dos elementos de formulário, deve-se utilizar os elementos do tipo label ou os atributos 'title'. Listamos, na tabela para download, cada função, o nome, o valor e o estado dos elementos em HTML.: estaciadocente.webaula.com.br/cursos/GRA270/doc/aula08_tabela.pdf

8.2 Dar aos usuários tempo suficiente para ler e para usar o conteúdo

Tempo Suficiente

Quando se trata de acessibilidade, deve-se pensar que cada usuário dispõe de um tempo determinado de resposta de acordo com a própria percepção ou com a própria condição física. Assim sendo, deve-se dar aos usuários dos sistemas e dos portais tempo suficiente para a leitura e para o uso do conteúdo dos mesmos.

8.3 Não utilizar conteúdos que provoquem convulsões

Crises

O conteúdo presente em um portal ou sistema não deve provocar convulsões, enjoos ou qualquer tipo de mal-estar devido à maneira como é apresentado. Imagens com trocas muito rápidas, luzes piscantes ou quaisquer recursos desse tipo devem ser evitados, pois podem desencadear esse tipo de problema naqueles que navegam pelo ambiente.

8.4 Ajudar os usuários a navegar e a encontrar o conteúdo

Navegáveis

Cabe ao desenvolvedor ajudar os usuários a navegar e encontrar o conteúdo que procuram dentro do portal em questão. Os mapas de sites e a estrutura de navegação com opções de mudança para páginas alternativas são recomendações fortes do WCAG.

No caso de SCRIPT, deve-se utilizar, para cada elemento script apresentado, o NOSCRIPT equivalente.

Código Exemplo

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

```
document.write("Olá Visitante!")
```

```
</script>
```

```
<noscript>O seu navegador não está com o Javascript ligado!</noscript>
```

Outro exemplo

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

```
document.write("Olá Visitante!")
```

</script>

<noscript>Vá direto para o conteúdo acessívelPágina Acessível</noscript>

9 Exemplos de funcionalidades acessíveis via teclado

9.1 Atalhos Via Teclado

Para se disponibilizar o acesso aos recursos das páginas via teclado, basta utilizar o atributo accesskey com o valor da letra a ser pressionada.

No caso do exemplo abaixo, ao se combinar as teclas ALT + SHIFT + C não se precisa navegar com o mouse.

No Internet Explorer, faz-se necessário, além da combinação de teclas, pressionar o ENTER para confirmar que se deseja navegar pelo link.

Código Exemplo:

```
<html>
<body>
<a accesskey="C" href="http://www.universidade.br" hreflang="pt-BR"
title="Universidade">
Universidade</a>
</body>
</html>
```

9.2 Acessando um formulário

Código Exemplo:

```
<FORM action="submit" method="post">
<LABEL for="usuario" accesskey="U">Nome</LABEL>
<INPUT type="text" id="usuario">
</FORM>
```

9.3 Longdesc

Ao se trabalhar com imagens, não ficam claras, pela descrição sucinta no atributo 'alt', todas as informações que poderiam estar disponíveis. Para que se disponibilizem maiores explicações, principalmente quando se trata de imagens que representam gráficos, deve-se utilizar o atributo 'longdesc' que indicará um arquivo externo contendo maiores explicações a respeito daquela imagem.

Código Exemplo:

```

```

Dentro do 'vendas_2010.html', tem-se um texto explicativo como no exemplo abaixo:

O gráfico de vendas de 2010, no primeiro trimestre, representa um aumento de 20% em comparação com igual período de 2009.

Percebe-se que, com isso, fica fácil poder utilizar bancos de dados para compor dinamicamente o conteúdo dos 'longdesc' de cada imagem arquivada nos sistemas dos usuários.

Ao se utilizar <OBJECT>, tem-se o mesmo efeito com o código:

```
<html>
```

```
<body>
```

```
<OBJECT data="vendas.jpg" type="image/jpeg">
```

O gráfico de vendas de 2010 no primeiro trimestre representa um aumento de 20% em comparação com igual período de 2009 Verifique nossos preços

```
</OBJECT>
```

```
</body>
```

```
</html>
```

9.4 Links

Neste exemplo, forneceram-se mecanismos para navegação utilizando o 'link' que provém de <a href>.

Não se esqueça que <a name>, isto é, a âncora, na mesma página, não fornece um função de 'link'.

- **Exemplo 1:**

No exemplo abaixo, o nome é o texto dentro do link, no nosso caso 'Google'.

Código Exemplo: Google

- **Exemplo 2:**

Neste exemplo, tem-se, dentro de um link, uma imagem cujo atributo 'alt' fornece o nome.

Código Exemplo:

- **Exemplo 3:**

A seguir, o exemplo indica o nome formado a partir da concatenação dos diferentes elementos dentro do link.

Código Exemplo:

```
<a href=" www.google.com.br ">Google</a>
```

- **Exemplo 4**

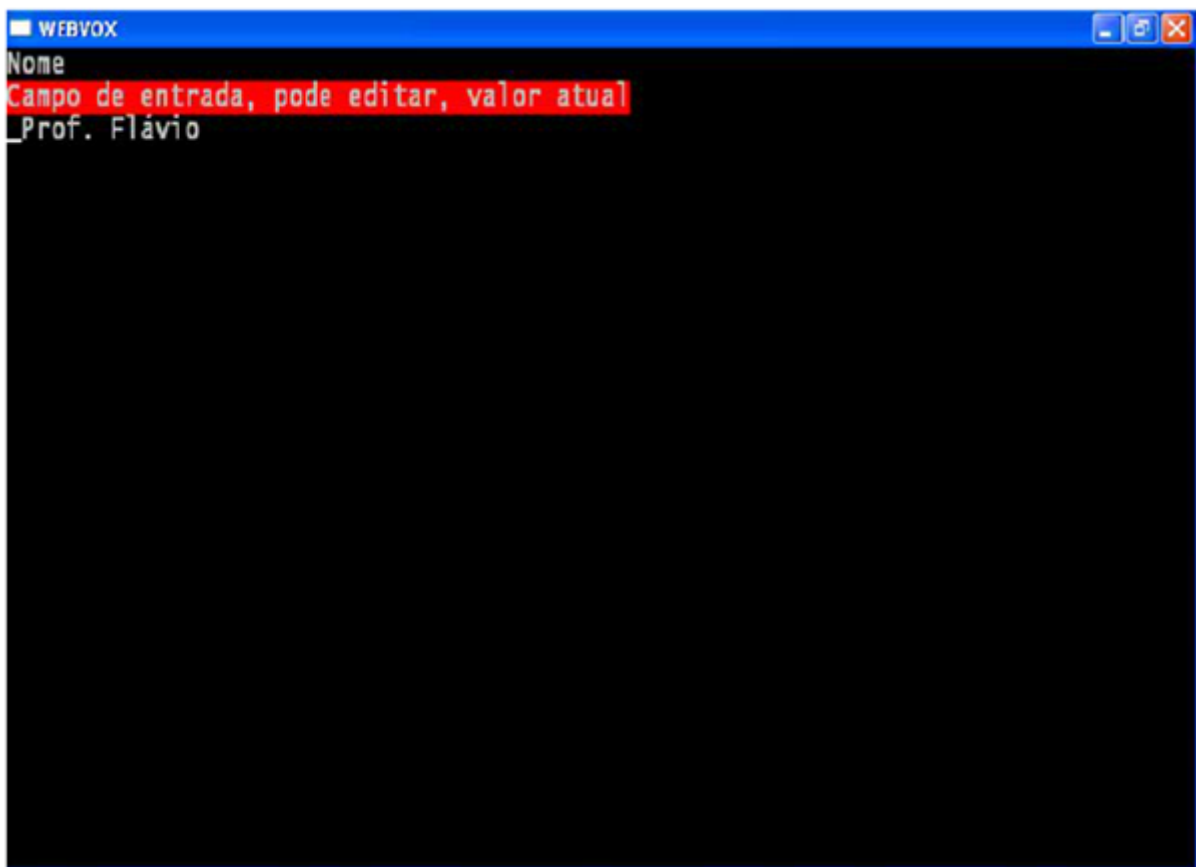
Neste exemplo, apresenta-se uma solução para o campo de entrada que tem uma função de texto editável. É importante que se observe que o elemento 'label' está associado ao elemento 'input' através do atributo 'for' que, por sua vez, faz referência ao mesmo através do atributo 'id' do elemento 'input'. O nome tem como origem o elemento 'label' e, neste caso, utilizou-se 'Nome'; 'Prof. Flávio' foi empregado como valor do atributo.

- **Exemplo 5**

Neste exemplo, o campo de entrada tem a mesma função e valor do exemplo anterior, mas obtém o respectivo nome a partir do atributo 'title'.

Código Exemplo:

```
<input id="text_1" type="text" value=" Prof. Flávio"  
title="Digite o seu primeiro nome">
```



9.5 Botões

Existem várias formas de se criar um botão em HTML e estão todas associadas à função 'botão'.

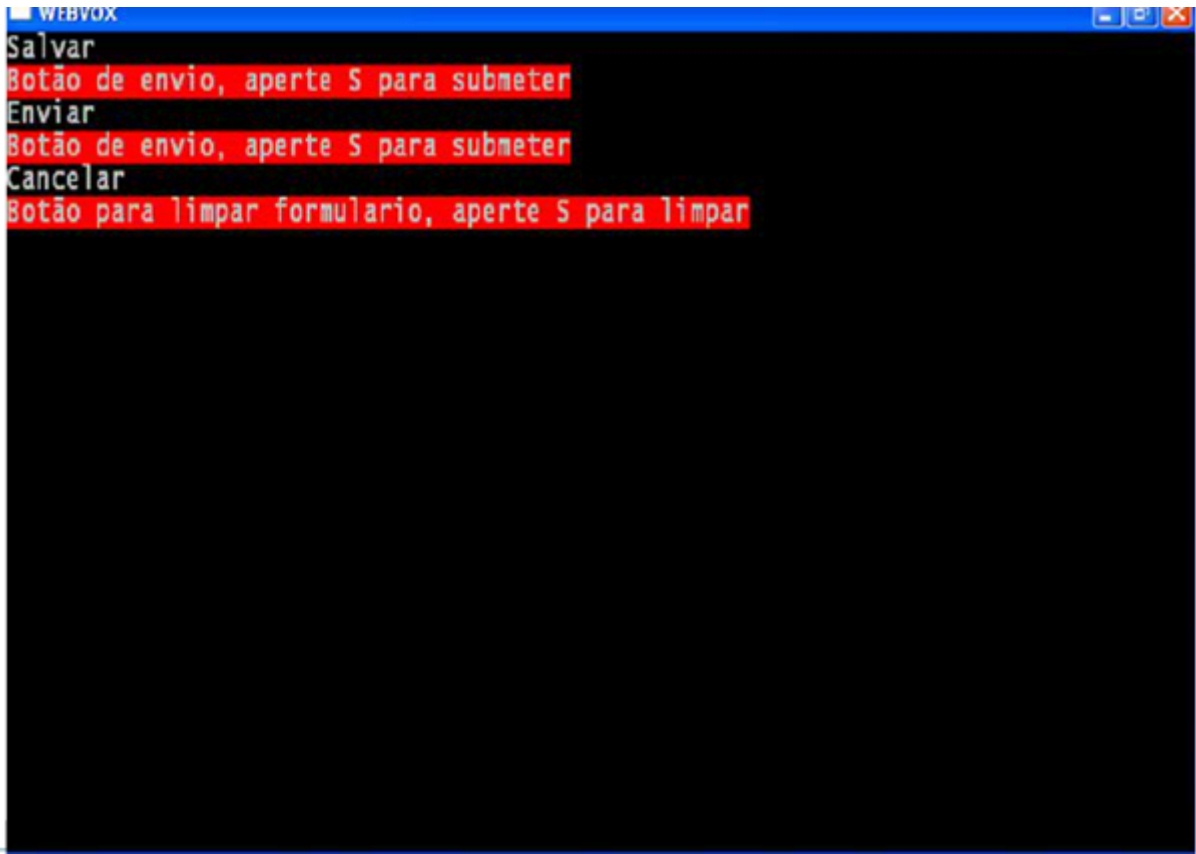
Exemplo 1

Neste exemplo, utilizou-se o atributo 'value' com o valor 'Salvar', 'Enviar' ou 'Cancelar', como o nome.

Código Exemplo:

```
<input type="button"  
value="Salvar" />  
<input type="submit"  
value="Enviar" />  
<input type="reset" value="Cancelar" />
```

Veja como isso fica no DOSVOX



9.6 Caixa de Verificação

Neste exemplo, tem-se uma caixa de verificação.

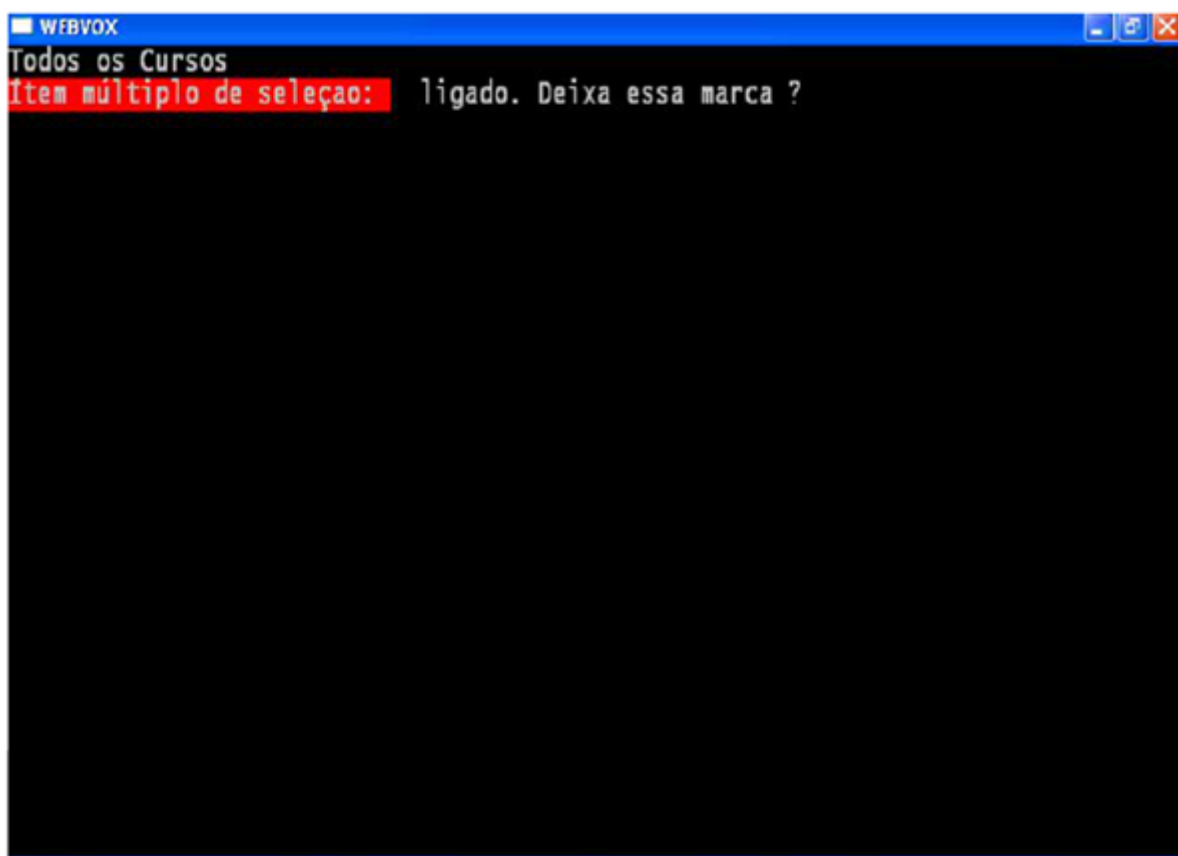
O elemento 'label' está associado ao elemento 'input' através do atributo 'for' que se refere ao atributo 'id' do elemento 'input'.

O estado pode ser 'Selecionado' ou 'Desmarcado' e provém do atributo 'checked'. Ao interagir, pode-se alterar este estado.

Código Exemplo:

```
<label for="cb_1">Todos os Cursos</label> <input id="cb_1" type="checkbox" checked="Selecionado">
```

Percebe-se, pela tela, que o DOSVOX apresenta o elemento marcado.



9.7 Botões de Opção

Foi utilizado, neste exemplo, os botões de opção identificados pelo atributo 'type' no elemento 'input'

O nome provém do elemento 'label'.

Deixou-se o item 'Análise de Sistemas' marcado.

Código Exemplo:

```
<input type="radio" name="curso" id="radio1" checked="checked"/><label for="radio1">Análise de Sistemas</label>
```

```
<input type="radio" name="curso" id="radio2" /><label for="radio2">Sistemas de Informação</label>
```

```
<input type="radio" name="curso" id="radio3" /><label for="radio3">Web Designer</label>
```

- **Exemplo 1**

Utilizou-se o elemento ‘caixa de combinação’ a ser identificado a partir do elemento ‘select’. O nome é ‘Cursos’ e é identificado pelo elemento ‘label’ que se relaciona com o elemento ‘option’ que, por sua vez, tem o atributo ‘selected’ definido como ‘selected’.

Código Exemplo:

```
<label for="s1">Cursos</label>
```

```
<select id="s1" size="1">
```

```
<option>Análise de Sistemas</option>
```

```
<option selected="selected">Sistemas de Informação</option>
```

```
<option>Web Designer</option>
```

```
</select>
```

- **Exemplo 2**

Este exemplo tem como variante o anterior, definindo o nome com o atributo ‘title’ no elemento ‘select’. Esta técnica pode ser utilizada quando não for pretendida uma etiqueta visível.

Código Exemplo:

```
<select id="s1" title="Cursos" size="1">
```

```
<option>Análise de Sistemas</option>
```

```
<option selected="selected">Sistemas de Informação</option>
```

```
<option>Web Designer</option>
```

```
</select>
```

O uso do ‘title’ para o DOSVOX não trouxe maiores informações ao usuário.

9.8 Textarea

- Exemplo 1

Neste exemplo, é analisado o comportamento do elemento 'textarea'.

A indicação do nome advém da etiqueta 'Digite aqui seu endereço' elemento 'label'.

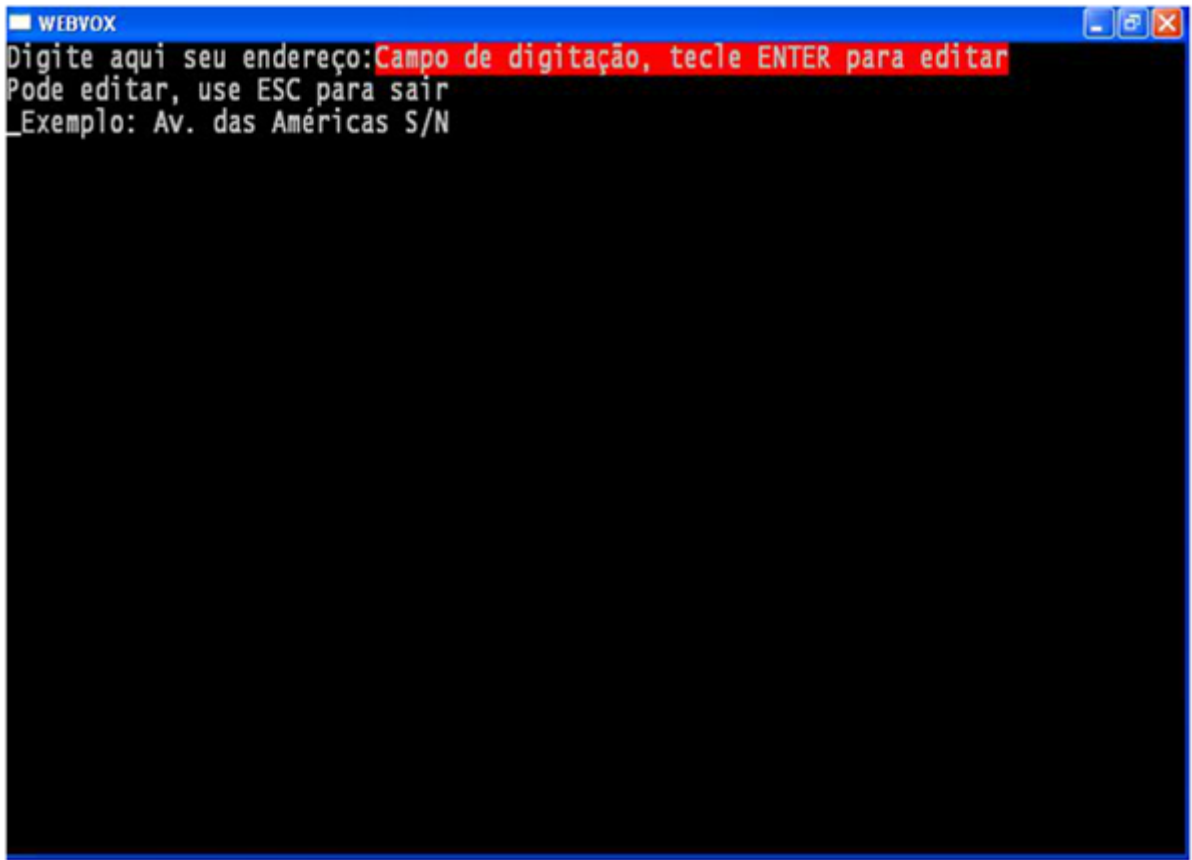
O valor é o conteúdo dentro do elemento 'textarea', neste caso 'Av. das Américas S/N'.

Código Exemplo:

```
<label for="ta_1"> Digite aqui seu endereço </label>
```

```
<textarea id="ta_1" >Exemplo: Av. das Américas S/N </textarea>
```

O DOSVOX coloca para o usuário um exemplo de endereço facilitando, assim, a compreensão do mesmo.



- Exemplo 2

Este exemplo tem a mesma função do anterior só que define o nome utilizando o atributo 'title'.

Código Exemplo:

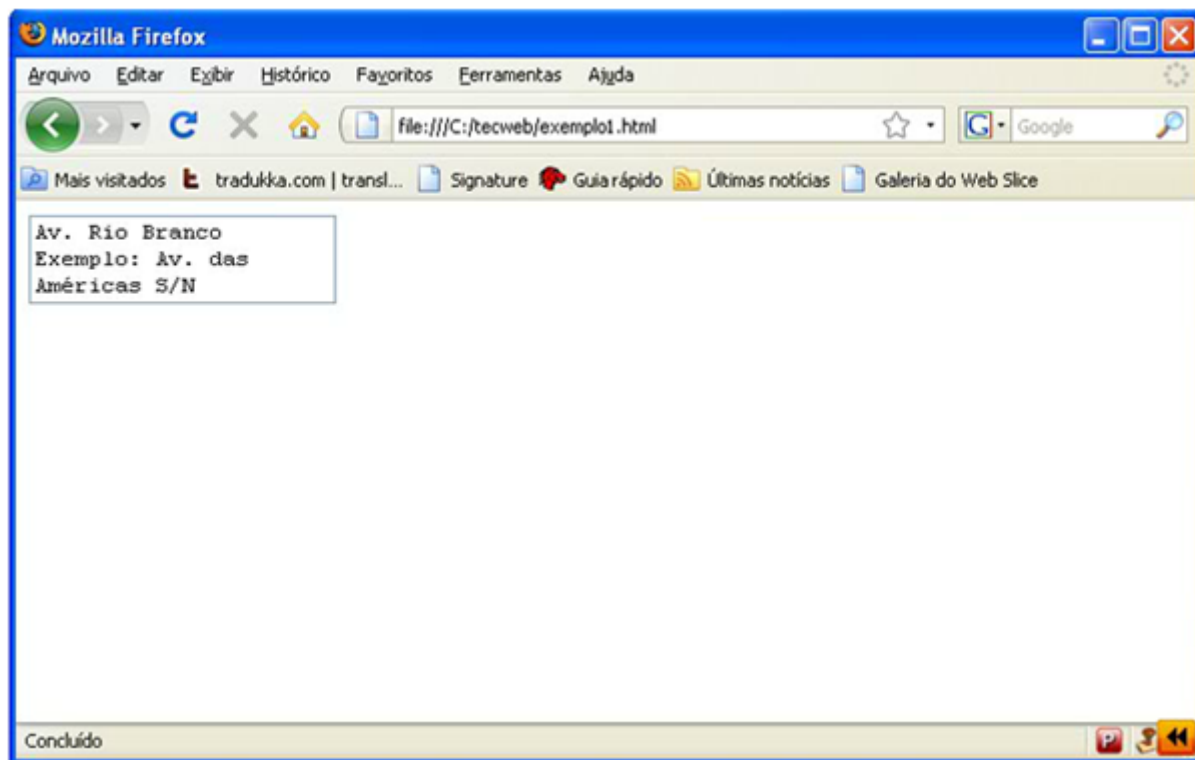
```
<textarea id="ta_1" title=" Digite aqui seu endereço " >
```

```
Exemplo: Av. das Américas S/N </textarea>
```

Neste caso, a troca do 'label' pelo 'title' não permitiu ao DOSVOX a identificação do elemento pelos recursos de voz.

Cuidado!!!

A utilização do recurso 'title' como identificador pode gerar efeitos não esperados quando se trata de usuários videntes, pois eles podem digitar a partir do seu conteúdo, concatenando-o com o valor descrito. A figura, ao lado, é um exemplo desse problema.



- **Exemplo 3**

O recurso do fieldset permite que sejam criados agrupamentos de forma elegante nos sistemas.

Através dele, podem-se definir grupos baseados em funções ou contextos específicos.

O nome provém do elemento 'legend'.

Código Exemplo:

```
<fieldset>
```

```
<legend>Escolha o Curso:</legend>
```

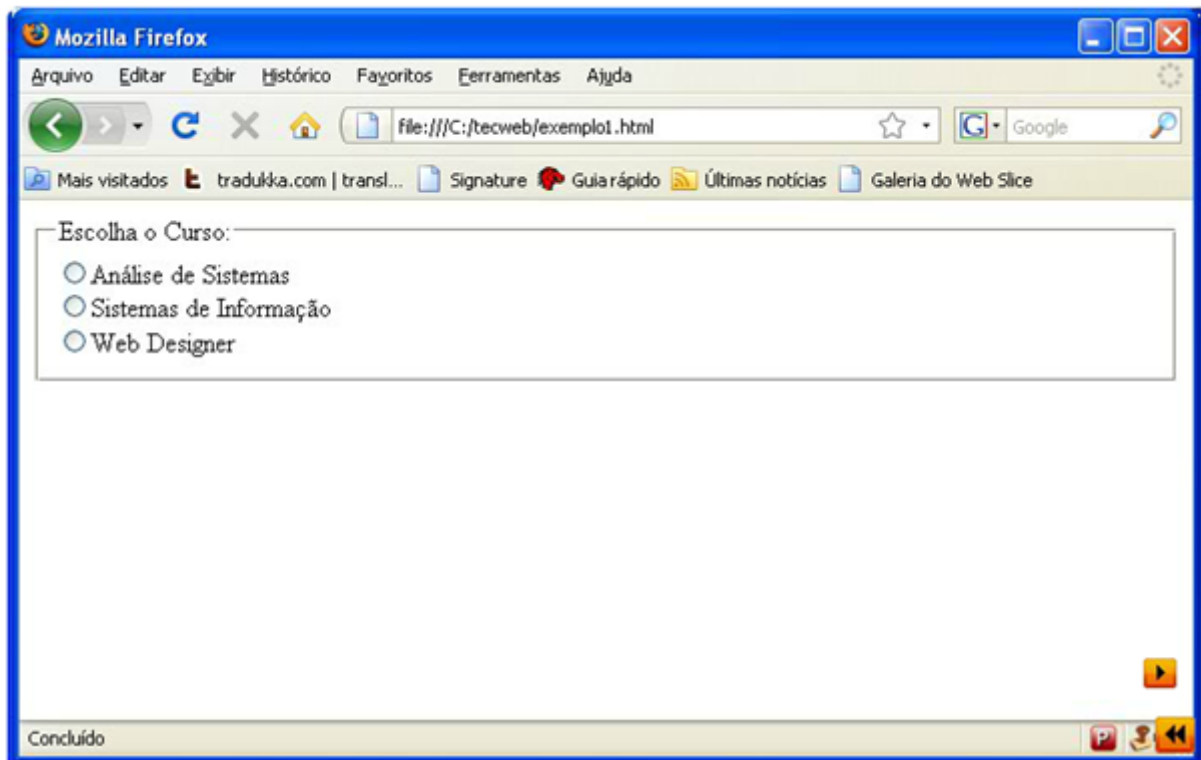
```
<input id="as" type="radio" name="curso" value="AN1" /><label for="as">Análise de Sistemas</label><br />
```

```
<input id="si" type="radio" name="curso" value="SI1" /><label for="si">Sistemas de Informação</label><br />
```

```
<input id="wd" type="radio" name="curso" value="WD1" /><label for="wd">Web Designer</label>
```

```
</fieldset>
```

O efeito visual deste recurso pode ser acompanhado na figura.



Utilizando navegação via teclado, utilizando 'tabindex' para a navegação via tecla TAB.

```
<fieldset>
```

```
<legend>Escolha o Curso:</legend>
```

```
<input id="as" type="radio" name="curso" value="AN1" tabindex="1"/>
```

```
<label for="as">Análise de Sistemas</label><br />
```

```
<input id="si" type="radio" name="curso" value="SI1" tabindex="2" />
```

```
<label for="si">Sistemas de Informação</label><br />
```

```
<input id="wd" type="radio" name="curso" value="WD1" tabindex="3"/>
```

```
<label for="wd">Web Designer</label>
```

```
</fieldset>
```