

Autoria Web

Aula 10 - Acessibilidade e manutenção

Apresentação

Na **Aula 9**, você aprendeu algumas técnicas de design para tornar os sites mais agradáveis e funcionais. Aprendeu também que a navegação é o ponto chave para tornar o conteúdo de um site acessível. Entretanto, acessibilidade não é só navegação.

Aqui, na nossa última aula, você aprenderá o que é acessibilidade e como tornar o seu site ainda mais acessível.

Além disso, você aprenderá que, para ter um site sempre interessante, é necessário atualizá-lo regularmente e, assim, conhecerá algumas dicas para tornar os documentos (X)HTML e CSS mais fáceis de atualizar e modificar.

Como esta é a última aula desta disciplina, falaremos um pouco sobre alguns tópicos que não foram abordados em aulas anteriores, mas que são fundamentais para um bom Webdesigner.



Vídeo 01 - Apresentação

Objetivos

Ao final desta aula, você deverá ser capaz de:

- Conceituar acessibilidade.
- Listar algumas maneiras de tornar um site mais acessível.
- Tornar os documentos (X)HTML e CSS mais fáceis de atualizar.

Acessibilidade

Ter um layout bonito, um texto bem escrito e uma boa navegação não são as únicas características necessárias em um site. Ele também deve ser acessível a todas as pessoas que desejem acessá-lo, inclusive pessoas que possuam algum tipo de necessidade especial. Se algum usuário não conseguir acessar o conteúdo de um site, ele procurará outro e não retornará, pois para ele essa página é **inacessível**.

Veja, a seguir, o que significa acessibilidade para Web:

“Acessibilidade para Web significa que pessoas portadoras de necessidades especiais sejam capazes de navegar na Web. Mais concretamente, significa uma Web projetada de modo a que essas pessoas possam: perceber, entender, navegar e interagir de uma maneira efetiva com a Web, bem como criar e contribuir com conteúdos para a Web”

Disponível em: <<http://maujob.com/w3c/introwac.html>>. Acesso em: 19 out. 2015.

Para começar a entender esse assunto, é necessário conhecer o conjunto de diretrizes de acessibilidade do conteúdo web (WCAG 2.0), definido pela W3C.

Existem as diretrizes de nível A, que são as exigências básicas, as de nível AA e as de nível AAA. Essa é uma lista geral de prioridades e está disponível em: <<http://www.ilearn.com.br/TR/WCAG20/>>. Acesso em: 19 out. 2015.

Você verá, nesta aula, algumas dicas para tornar seu site acessível. Mas, antes disso, vamos ver um vídeo?

Neste vídeo, você verá alguns exemplos das dificuldades encontradas pelos usuários e algumas formas simples de resolvê-las. Acesse e assista.

<<http://www.youtube.com/watch?v=hFI4CuxQjSA>>. Acesso em: 19 out. 2015.

Certamente, esse vídeo esclareceu o quão problemático é um site mal projetado, no que diz respeito à acessibilidade, não?!

Para iniciar nosso entendimento sobre acessibilidade, pensaremos em alguém com necessidades visuais acessando um site. Para ele, não será importante se o site tem uma boa aparência, ou se tem cores bonitas, pois ele o perceberá de outra maneira.

Como você viu no vídeo, a maioria dos deficientes visuais utiliza um leitor de tela, ou seja, um programa projetado para ler a página para o usuário. Esse tipo de programa lê o texto do site em voz alta, com base na marcação (X)HTML. Sendo assim, ele informa quando está lendo um link, um cabeçalho (detalhando seu nível) etc. Um dos leitores de telas mais populares é o JAWS.

Que tal fazer um teste? Faça o download do JAWS. Disponível em: <<http://www.freedomscientific.com/downloads/jaws/jaws-downloads.asp>>. Acesso em: 19 out. 2015, instale-o e tente ler um site com esse leitor para experimentar como um deficiente visual percebe uma página web.

Para fazer o JAWS ler um site, você deve executá-lo e abrir seu browser. Insira um endereço na barra de endereços do navegador. Quando a página estiver carregada, você deve utilizar a tecla tab, para que o JAWS leia elemento por elemento clicável, ou usar as setas para baixo e para cima para que ele leia tudo de cada linha <<http://maujor.com/tutorial/usando-jaws-para-testes.php#intro>>. Acesso em: 19 out. 2015.

A acessibilidade Web envolve não só as deficiências visuais, mas também motoras, auditivas, cognitivas e neurológicas, como também as limitações das pessoas com mais idade.

Dessa forma, inclui providências tais como mudanças no tamanho e cor da fonte, planejamento da ordem de marcação, uso de alguns atributos e tags, dentre outros.

A seguir, você verá que simples cuidados farão com que pessoas com limitações desfrutem do seu site.

No site <<http://www.ilearn.com.br/TR/WCAG20/>>. Acesso em: 19 out. 2015., são definidas 4 recomendações gerais para prover acessibilidade a documentos web e indicações de como essas recomendações podem ser implementadas.

Recomendação 1 - “Perceptível: A informação e os componentes da interface do usuário têm de ser apresentados aos usuários em formas que eles possam perceber”.

Na aula passada, você viu o quanto é importante que as informações sejam realçadas com cores e tamanhos, e que os links pareçam realmente clicáveis. Isso é importante para as pessoas que conseguem ver. Para as que não conseguem ver, tudo isso terá que ser repensado.

Como você assistiu no vídeo, os programas usados por essas pessoas para ler um site se baseiam na marcação (X)HTML. Assim, uma imagem é só um nome de um arquivo com uma extensão (caso o atributo alt do elemento img não for definido apropriadamente), um link é o texto do link mais a palavra “link”, e assim sucessivamente.

Por exemplo, a marcação é lida da seguinte maneira: “imagem1 ponto jpg”. Você sabe que não é assim que uma pessoa deseja perceber uma imagem, ela precisa saber qual o conteúdo da imagem, o que ela representa.

Sendo assim, a primeira coisa a fazer é fornecer um texto aos elementos não textuais, tais como imagens, vídeos, sons, applets, dentre outros (essa é uma diretriz de nível A).

Para figuras, isso pode ser feito da seguinte maneira:

- Incluindo o atributo **alt** – nesse caso, os leitores de tela, em vez de lerem o nome do arquivo, lerão o texto indicado nesse atributo. Você deve usar o atributo alt para descrever **brevemente** o conteúdo do elemento não textual. Caso você deseje oferecer uma descrição mais detalhada, é melhor usar o atributo longdesc.
- Incluindo o atributo **longdesc** – nesse caso, os leitores de tela darão ao usuário a possibilidade de acessar uma página que conterá uma descrição mais **longa e detalhada** sobre o elemento não textual. Veja o exemplo abaixo:

```
1 
2 ...
3 <body>
4   <p>Linda paisagem do parque chapada diamantina. </p>
5 </body>
6 </html>
```



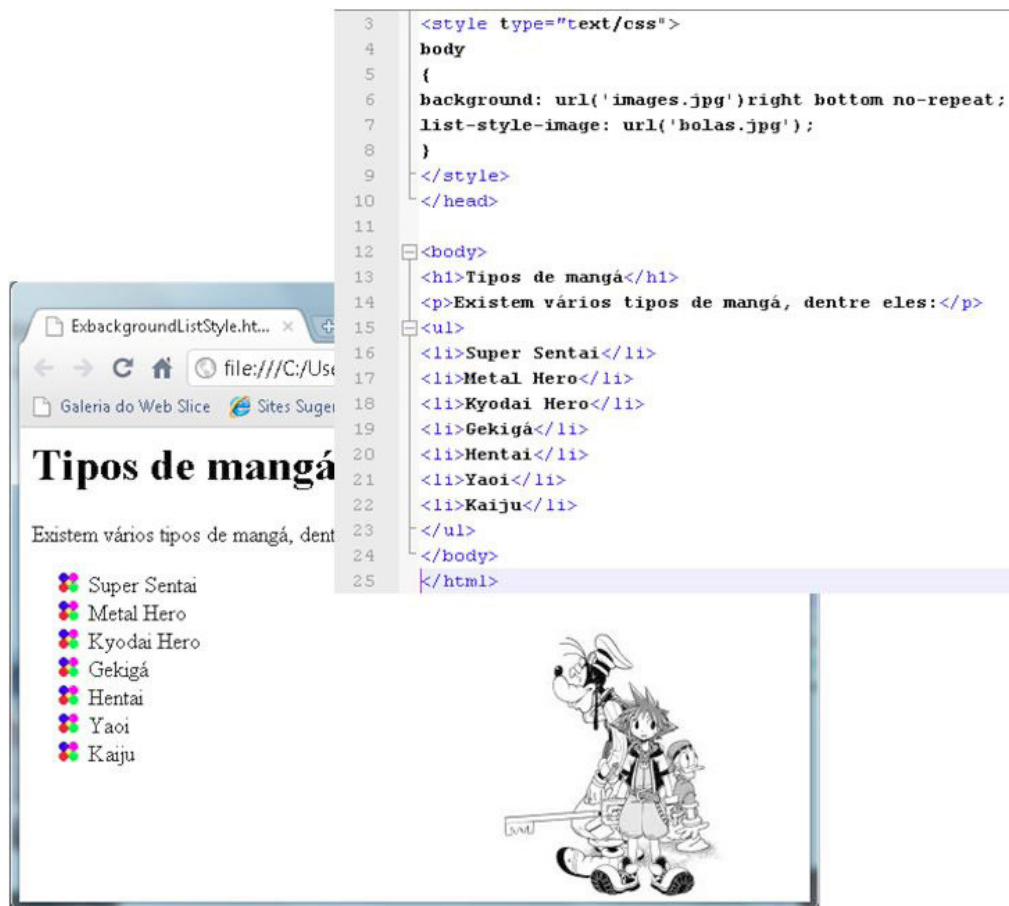
Vídeo 02 - Acessibilidade

Além de adicionar textos para representar as imagens, é necessário também analisar se as figuras existentes no (X)HTML são puramente decorativas ou não.

Para figuras puramente decorativas, ao invés de adicionar texto referente a elas, é melhor excluí-las do (X)HTML e inseri-las somente através do CSS. Afinal de contas, o objetivo do CSS é exatamente esse, estilizar e decorar o documento (X)HTML, enquanto que o objetivo do (X)HTML é somente especificar conteúdo.

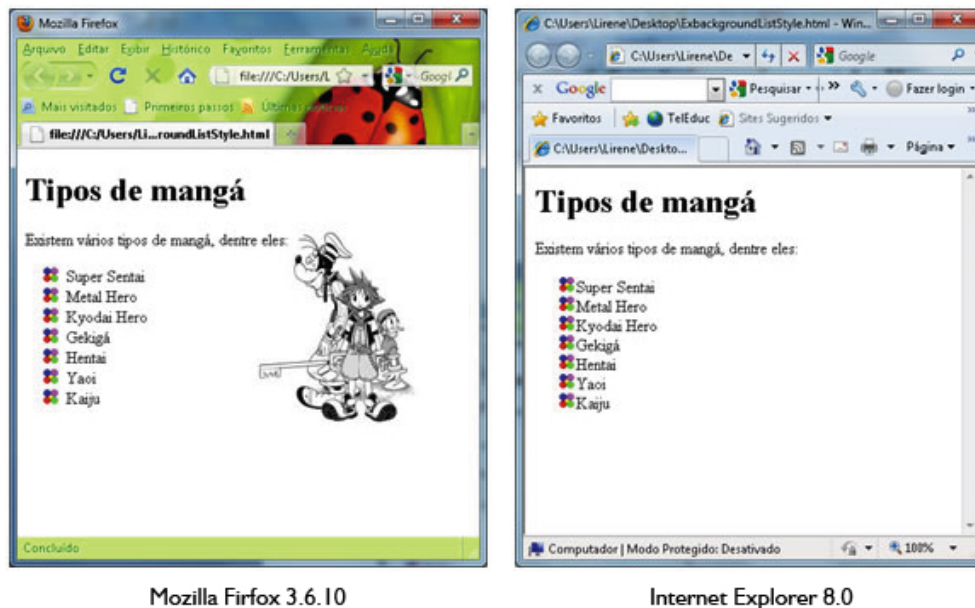
Quanto melhor essa separação entre conteúdo e estilo, melhor para acessibilidade e para a manutenção das páginas. Sendo assim, você pode usar o atributo background e list-style-image, respectivamente, para adicionar uma imagem ao plano de fundo da página (ou de um elemento) e para estilizar o marcador dos itens de uma lista. Veja um exemplo na **Figura 2**.

Figura 02 - Exemplo do uso dos atributos background e list-style-image



Infelizmente, nem todos os browsers se comportam da mesma maneira. Veja, na **Figura 3**, como esse mesmo html é apresentado no Mozilla e no IE8. No caso do Google Chrome, a figura é posicionada o mais à direita e abaixo possível, e isso varia de acordo com o tamanho da janela. Enquanto no Mozilla, a figura é posicionada abaixo de todo o conteúdo, independente do tamanho da janela. No IE8, a figura nem é mostrada.

Figura 03 - Exemplo de erro na apresentação



Vídeo 03 - Diretrizes

Infelizmente, enquanto os browsers não seguem as diretrizes gerais indicadas pela W3C, é papel do designer corrigir esses problemas. Mas, a boa notícia é que isso mudará com o tempo.

Outra diretriz diz: “A cor não é utilizada como o único meio visual de transmitir informações, de indicar uma ação, de pedir uma resposta ou de distinguir um elemento visual (Nível A)”.

Sendo assim, é necessário garantir que toda informação em cores também exista sem cores, pois alguns usuários possuem uma falta de sensibilidade a certas cores, como, por exemplo, verde e vermelho.

Você aprendeu, na aula passada, que, quando desejamos melhorar a informação de uma imagem (ícone), adicionamos texto a ela. Analogamente, podemos realizar esse mesmo procedimento aqui.

Podemos escolher comunicar uma informação simultaneamente com cores e texto. Essa providência tornará o site acessível para os usuários com dificuldade de visualização de cores.

Para você compreender melhor as dificuldades de um usuário com esses problemas, veja as figuras abaixo, que mostram, respectivamente, uma paleta de cores e como ela é vista por um usuário [daltônico](#) (Daltonismo: perturbação da percepção visual em que se vê várias cores, caracterizada pela incapacidade de diferenciar todas ou algumas cores, manifestando-se muitas vezes pela dificuldade em distinguir o verde do vermelho. (Fonte: <http://pt.wikipedia.org/wiki/Daltonismo>)).

Figura 04 - Paleta de Cores como vista por pessoas com deuteranopia, uma forma comum de daltonismo



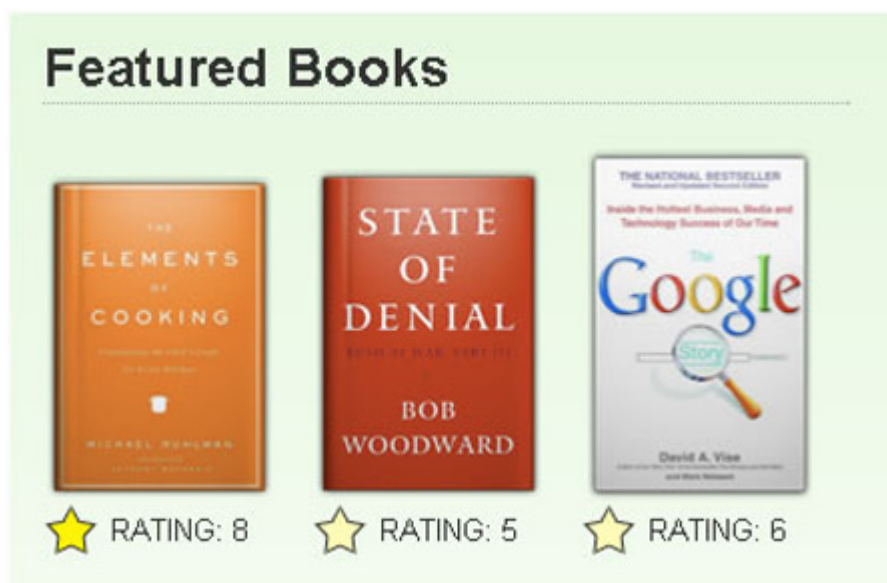
Fonte: [Microsoft](#). Acesso em: 19 out. 2015.

Viu a diferença? As duas paletas parecem ser completamente diferentes. .

Você não sabe qual paleta seu usuário verá. Portanto, quando usar cores para representar informação extra (além da escrita), adicione texto para comunicar a mesma informação.

Veja o exemplo mostrado na **Figura 5**. A cor da estrela indica o quão boa é a avaliação do livro, mas a mesma informação representada pela intensidade da cor é representada também pelo texto que a acompanha.

Figura 05 - Exemplo de uso de texto para fornecer informação representada por cores



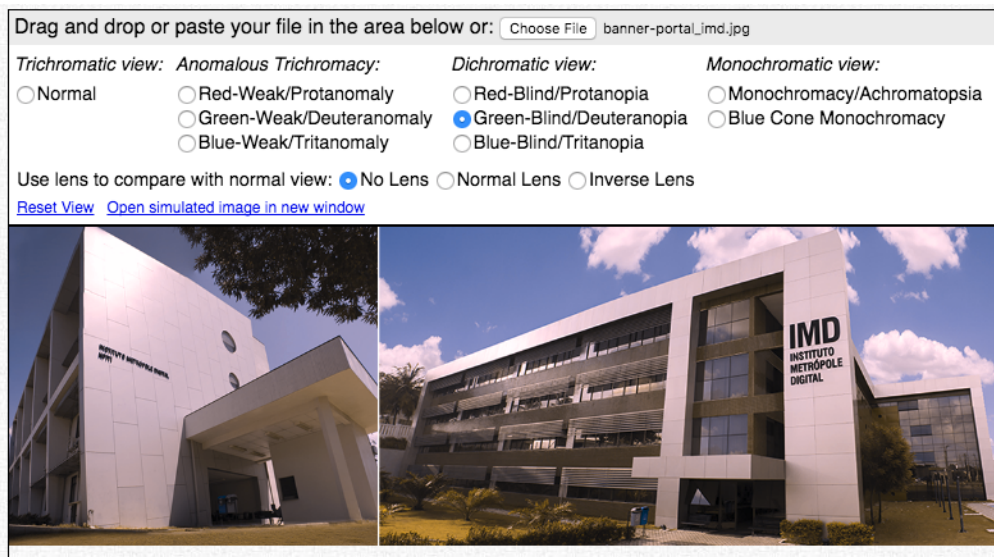
Fonte: Watrall e Siarto(2009)

Existem alguns serviços on-line e gratuitos que podem simular a aparência de uma imagem ou mesmo de seu site para pessoas com algumas variações de daltonismo. Um deles é o Coblis. Para utilizá-lo, basta entrar no endereço <<http://www.color-blindness.com/coblis-color-blindness-simulator/>> e seguir o roteiro seguinte:

Para simular com uma imagem:

- Onde tem o texto “Drag and drop or paste your file in the area below or:” Clique na opção de escolher uma imagem do seu computador.
- Após escolher a imagem ela deverá aparecer no site. Escolha uma das opções de simulação, digamos “Green-Blind/Deuteranopia” como visto na **Figura 6**.

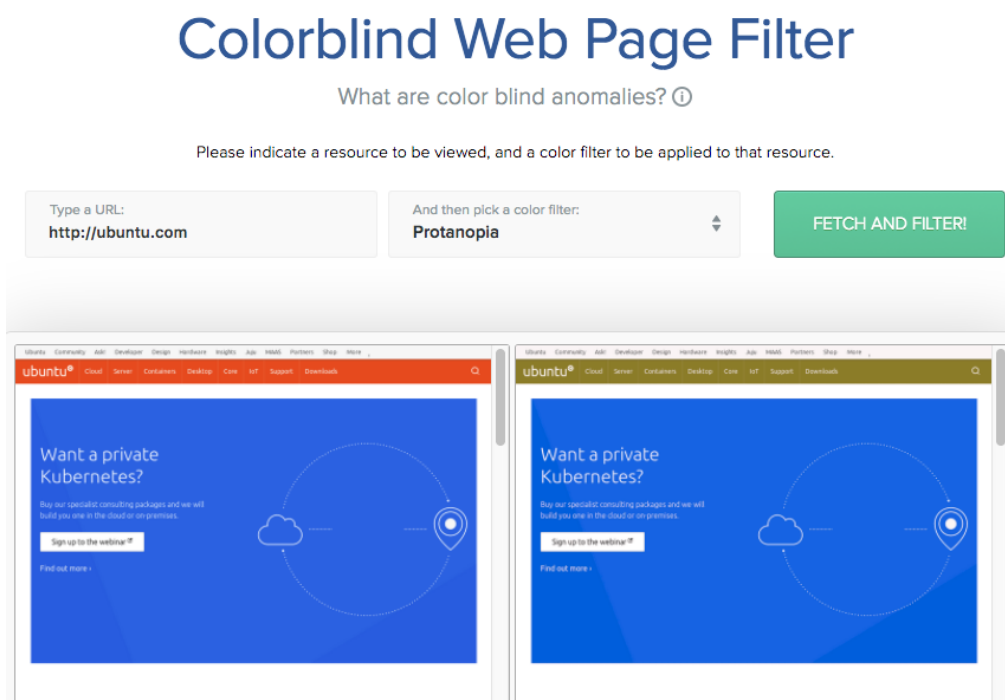
Figura 06 - Imagem com filtro de cores aplicado simulando a visão da mesma por pessoas com perturbação de percepção visual.



Repare que cada opção marcada simula a visão de alguém com a condição escolhida. A opção “Normal” faz com que a imagem seja exibida da forma original

Outra opção interessante para se testar um site inteiro é o [Colorfilter](#). Nele você digita a URL de um site e é exibida sua versão simulada com as cores vistas por alguém com perturbação de percepção visual. Ver **Figura 7**.

Figura 07 - Site Colorfilter exibindo um site inteiro com cores simulando a visão de pessoas com perturbação de percepção visual.



O Colorfilter nem sempre funciona com todos os sites, então é sempre mais garantido realizar os testes com imagens individuais, como visto anteriormente.

Recomendação 2- “Operável: Os componentes de interface de usuário e a navegação têm de ser operáveis”.

Essa recomendação está associada à facilidade de acessar e navegar no conteúdo.

Uma das diretrizes é que “haja um mecanismo para ignorar blocos de conteúdo que são repetidos em várias páginas Web (Nível A)”.

Assim, uma das formas de implementação é através da adição de um link no topo da página que leve o usuário para o conteúdo principal do site.

Como você viu no vídeo, isso facilita a navegação porque os usuários que utilizam os programas de leitura de tela não precisam escutar referência a todos os links e informações secundárias do topo da página. Eles podem simplesmente acessar direto o link referente ao conteúdo principal.

Esse link pode ser um link visível para todos ou pode ser um link oculto que você pode estilizar com a propriedade de posicionamento, de tal forma que ele não apareça no browser, como no exemplo abaixo, através da propriedade top com valor negativo.

```
1 #linkPrincipal
2 {
3   position:absolute;
4   display:block;
5   top:-100;
6 }
```

Outro ponto chave é que deve-se garantir a acessibilidade do conteúdo via teclado.

Isso beneficia não apenas deficientes visuais, mas também pessoas com deficiências motoras e pessoas que não têm acesso ao mouse (por exemplo, uma pessoa acessando um site via telefone celular).

De maneira geral, o usuário que usa um leitor de tela, percorre um site de uma das duas formas:

- Clicando a tecla tab — nesse caso, ele percorre o site indo de link em link;
- Clicando as setas para baixo, para cima para ler a linha inteira (ou clicando insert+seta para baixo para que leia-se tudo).

Como a ordem dos elementos lidos é a ordem em que eles estão no (X)HTML e, como vimos, é possível posicionar os elementos à esquerda ou à direita com CSS, então, é possível que o usuário escute o documento não exatamente na mesma ordem em que aparece na tela (a ordem planejada pelo webdesigner). Isso algumas vezes pode levar o usuário a se perder, pois nem tudo que ele escutar estará realmente contextualizado da maneira correta.

Para ordenar o conteúdo de forma significativa e amenizar esse problema, pode-se fazer duas coisas:

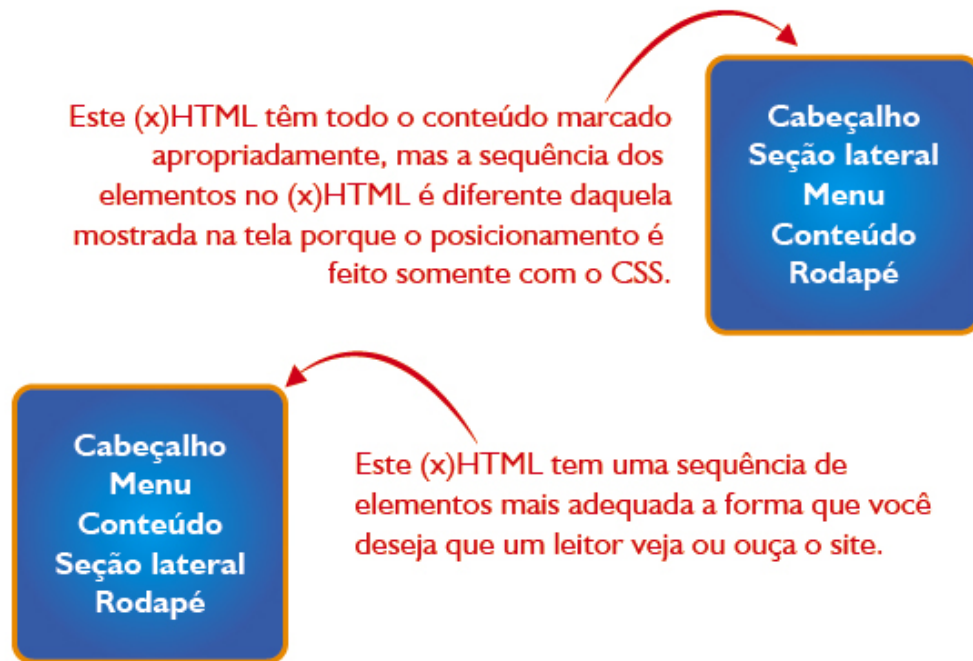
1 – Escrever todo o conteúdo na ordem em que realmente deve ser lido — A primeira coisa a fazer é tentar, o máximo possível, escrever o conteúdo no (X)HTML na ordem em que ele deve ser lido.

Por exemplo, na **Aula 8**, você aprendeu a usar o elemento div para dividir o documento em várias partes, tais como as mostradas na **Figura 8** (parte superior).

Imagine uma situação em que essas seções são escritas no (X)HTML na seguinte ordem: cabeçalho, seção lateral, menu, rodapé e conteúdo. Talvez o autor tenha pensado nessa ordem porque ele concentraria as menores seções juntas e deixaria a maior delas por último, ou por qualquer outro motivo.

Nesse caso, a ordem em que os elementos aparecem na tela é estabelecida somente pelo CSS, e isso é satisfatório para um internauta sem necessidades especiais; mas, para aqueles que usam um leitor de tela ou que usam somente o teclado como meio de interação, será um problema.

Figura 08 - Exemplo de ordem do conteúdo no (X)HTML



Por que não escrever o documento com uma sequência que esteja mais de acordo com aquela estilizada pelo CSS?

Veja o segundo exemplo da **Figura 8** (parte inferior). Se o usuário usar a tecla tab ou ler a tela através das setinhas, ele navegará no conteúdo em uma sequência mais parecida com a que você deseja que ele perceba o site.

Assim, planeje a ordem do conteúdo do XHTML sem o CSS. Isso te ajudará a posicionar os elementos na ordem em que você quer que alguém navegue no site.

Essa simples mudança da ordem vale não somente para os elementos divs, mas para todos os elementos e pode trazer muitos benefícios para pessoas com necessidades especiais.

2 - Estabelecer a ordem em que os links devem ser lidos através do atributo tabindex — Uma segunda opção para ordenar a sequência em que os links serão lidos é usar o atributo tabindex no elemento `` (ou elementos de um formulário).

Esse atributo fornece a posição na ordem de tabulação dos elementos. Seu valor é preenchido com um número. Por exemplo, se um elemento é marcado com um tabindex de valor "3", ele virá antes do elemento marcado com o tabindex "4" e depois do elemento marcado com o tabindex "2".

Assim, se você, por algum motivo, não conseguir organizar a ordem do conteúdo como sugerimos acima, você pode usar o tabindex para definir essa sequência.

Nesse caso, você deve iniciar a marcação com tabindex pelo início da página: cabeçalho, navegação de alto nível, conteúdo principal e assim por diante. Isso permitirá que o usuário tenha acesso inicialmente ao menu principal que regula toda a navegação do site.

Mas, atenção! Aconselhamos que você dê prioridade à primeira opção (ordenar o conteúdo da maneira apropriada), pois ela garante que tanto um leitor de tela quanto a navegação por tab sejam realizados na sequência mais apropriada, enquanto o uso de tabindex só favorece a navegação através de tab.

Além disso, você perceberá que manter a numeração do tabindex é muito mais trabalhoso que tentar organizar a ordem real do conteúdo.

Recomendação 3 – “Compreensível: a informação e a operação da interface de usuário têm de ser compreensíveis”

Uma das diretrizes é que o idioma no qual o site é escrito deve ser determinado programaticamente de forma que as ferramentas possam identificá-lo (Nível A).

Uma forma de fazer isso é usar o atributo lang na tag <html> para indicar em que idioma o site é escrito.

Ao ler essa informação, leitores de tela tal como o JAWS (versão 8.0) conseguem mudar o idioma falado para o idioma definido no atributo lang, automaticamente.

```
1 <html lang="pt">
```

Esse atributo também pode ser usado na tag , caso haja palavras em diferentes idiomas no meio de uma frase. Por exemplo, se a palavra "site" estiver em um site escrito em português, o leitor de tela lerá "site". Se você marcar essa palavra com ..., então será lida como "saite".


```
1 <html lang="pt">
2 ...
3 <body>
4   <p> o <span lang="en">site</span> é especificado por elementos e ....</p>
5   ...
6 </body>
7 </html>
```

Recomendação 4- “Robusto: O conteúdo tem de ser robusto o suficiente para poder ser interpretado de forma concisa por diversos agentes do usuário (browsers), incluindo tecnologias assistivas.”

Há uma diretriz que diz: “No conteúdo implementado, utilizando linguagens de marcação, os elementos dispõem de marcas de início e de fim completas, os elementos estão encaixados de acordo com as respectivas especificações, os elementos não contêm atributos duplicados e todos os IDs são exclusivos, exceto quando as especificações permitem essas características (Nível A).”

Essa diretriz está em consonância com a utilização de (X)HTML. Assim, cumprir as regras de (X)HTML e garantir que as páginas sejam bem formadas e que os elementos (X)HTML sejam usados da maneira apropriada (por exemplo, o elemento h1 realmente marca um título de seção de nível 1) é um passo para o cumprimento dessa diretriz.

Deve-se ter atenção para não usar elementos deprecated e, cuidadosamente, escrever conteúdo que leve a tecnologias que não dão suporte à acessibilidade, ou que podem ser ligadas e desligadas tais como javascript e flash.

Agora, você conhece alguns dos critérios para que uma página seja considerada acessível.

Entretanto, para que uma página Web esteja em conformidade com a versão WCAG 2.0, devem ser cumpridos vários outros critérios. Acesse <<http://www.ilearn.com.br/TR/WCAG20/#intro-related-docs>>. Acesso em: 19 out. 2015 para conhecer todos eles e começar a tomar as providências necessárias para tornar a Web mais acessível.

Existem algumas ferramentas disponíveis para avaliar a acessibilidade de sites, tais como [Hera](#), [TAW](#), [DaSilva](#), [CynthiaSays](#) ou [wave](#) (Acesso em: 19 out. 2015). Veja nas **Figuras 9 e 10** os resultados dados pelas ferramentas wave e DaSilva para o site

do Gmail, respectivamente.

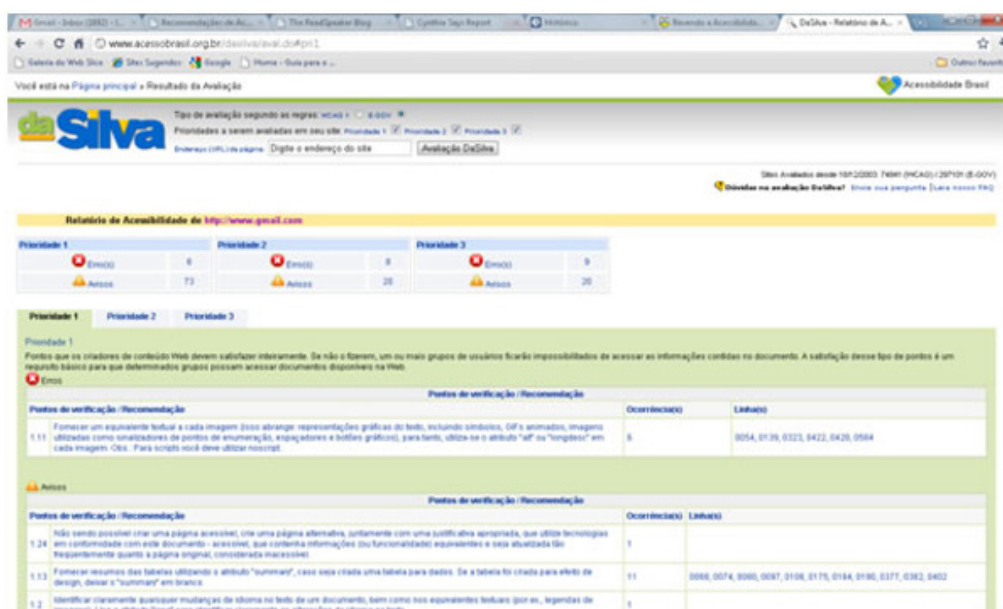
No resultado do wave, mostrado na Figura 9, os ícones representam alertas ou erros, que são descritos ao passar o mouse sobre eles.

Figura 09 - Exemplo de avaliação realizado pela ferramenta wave



Fonte: Primeira versão do curso

Figura 10 - Exemplo de avaliação realizado pela ferramenta DaSilva



Fonte: Primeira versão do curso

Que tal testar as páginas que você desenvolveu?

Você pode solicitar ao wave que a avaliação de um arquivo seja realizada, basta escolher o arquivo através do formulário disponível em <<http://wave.webaim.org/>>.



Vídeo 04 - Acessibilidade HTML

Atividade 01

1. Explique, com suas palavras, o que é acessibilidade.
2. O que é a WCAG 2.0?
3. Cite duas formas, diferentes das descritas nesta aula, de tornar um site mais acessível (ou menos inacessível).

Introdução à Manutenção

Temos falado, durante a disciplina inteira, em como tornar as páginas mais bonitas e úteis para todas as pessoas.

E o que há por trás desses adjetivos?

O que é bonito hoje, pode parecer ultrapassado amanhã. A informação útil hoje deve ser atualizada amanhã.

Enfim, de que adianta um site bonito e acessível se ele não muda? Um site que não mostra novidades, seja de conteúdo ou de estilo, se tornará cada vez menos útil e menos interessante com o passar do tempo. Assim, um ponto essencial a ser tratado é a manutenção das páginas desenvolvidas.

Manutenção

Talvez por algum momento, você tenha pensado “enfim, o site está perfeito!”.

Nada disso! No dia seguinte, você perceberá que os concorrentes estão usando uma nova forma de apresentação, o público-alvo tem uma nova demanda ou o público-alvo mudou.

Tudo na internet é muito rápido. Tudo se multiplica muito rápido e, se você não estiver antenado, logo, logo, o site que você acabou de desenvolver terá ficado ultrapassado. Assim, a necessidade de mudar é real.

Mudanças dão a ideia de evolução, progressão, domínio do mercado. Isso não significa que você deve criar um site completamente novo o tempo inteiro. Significa, sim, que você deve avaliar as melhores ideias, principalmente aquelas que se encaixam com o perfil de seu público-alvo, e adicioná-las ao site, por exemplo, adicionar um blog, uma nova forma de navegação ou uma metáfora visual diferente.

Tipos de Mudança

De maneira geral, há dois tipos de mudança: no conteúdo e no layout. Assim, quanto mais separados estiverem o conteúdo e o estilo, mais fácil será manter o site atualizado.

Então, a primeira providência a tomar é certificar-se, mais uma vez, de que o conteúdo está marcado apropriadamente pelo XHTML e de que toda a estilização é realizada através do CSS.

Lembre-se de que você pode usar as ferramentas de validação de XHTML e CSS disponíveis pela W3C para averiguar se os documentos estão corretos.

Quanto ao conteúdo, é necessário prover novidades, sejam relacionadas a conceitos ou à forma de interação, e o estilo deve acompanhar essas mudanças.

Além disso, você pode reorganizar o CSS para torná-lo mais fácil de mudar. Para isso, você deve analisar o CSS e identificar os elementos que possuem o mesmo estilo e, assim, agrupar tais elementos em uma mesma regra.

Por exemplo, se duas listas identificadas com ids usam a mesma cor de plano de fundo, dê preferência por escrever como mostrado na **Figura 11** à esquerda a escrever como mostrado na **Figura 11** à direita. Embora ambos estejam corretos, será mais fácil modificar o primeiro do que o segundo.

Figura 11 - Exemplo de melhoria na escrita do CSS

<pre>ul#lista1, ul#lista2{ background: gray; }</pre>	<pre>ul#lista1{ background: gray; } ul#lista2{ background: gray; }</pre>
--	--

Além disso, você pode pensar em adicionar interatividade ou dinamicidade ao site, seja por meio de javascript, flash ou outras linguagens de programação; mas lembre-se de que cada uma dessas tecnologias possui prós e contras.

Enfim, você chegou à última aula deste curso e, certamente, estará entusiasmado demais para pensar que acabou.

Com certeza não acabou, esse curso abordou apenas uma pequena parte, a mais básica, do que você precisa saber para trabalhar como webdesigner. A seguir, são descritas algumas das atividades exercidas por esses profissionais.



Vídeo 05 - Dicas de Manutenção

Atividades de um Webdesigner

Em Watrall e Siarto (2009), são listadas algumas das principais atividades desempenhadas por um webdesigner, descritas a seguir:

Designer de front-end – está relacionado à especificação HTML e CSS, deve-se ter uma noção de programação e, ao mesmo tempo, saber dispor os elementos harmonicamente na tela. Essa atividade consiste em grande parte de elaborar a interface com o usuário.

Redator – está relacionado a desenvolver e escrever o conteúdo, por exemplo, selecionando as melhores palavras para serem usadas nos links e tornando o texto mais enxuto e interessante para os leitores da web.

Designer gráfico – está relacionado à elaboração de um trabalho de arte para construir a metáfora do site, normalmente utiliza ferramentas de edição gráfica e trabalha junto ao designer de front-end para o desenvolvimento do layout.

Arquiteto da informação – está relacionado à elaboração e organização do conteúdo do site, bem como de sua navegação.

Designer da experiência do usuário – está relacionado a elaborar uma interface consistente e útil, além de gerenciar os testes de utilização dessa interface, levando em consideração a usabilidade e acessibilidade.

Programador – está relacionado a escrever códigos para tornar o site dinâmico, ou para tratar formulários, ou mesmo para gerar o conteúdo estático dinamicamente. Além disso, cresce a necessidade de oferecer serviços via web, tais como: venda de produtos, transações bancárias, dentre outros.

Como você pode ver, o leque de oportunidades é grande. E nem sempre você precisa ser especialista em todas essas atividades, pois, em muitas empresas de desenvolvimento, essas atividades são desempenhadas por pessoas diferentes, assim elas podem focar e se aprimorar em um só tipo de trabalho.

No entanto, para um iniciante, é melhor conhecer pelo menos um pouco de cada uma dessas áreas, até mesmo para que se possa escolher a que mais tem afinidade com você. O curso abordou, por alto, alguns assuntos relacionados a essas atividades, tais como layout, navegação, arquitetura da informação, acessibilidade, dentre outros. Mas, para acompanhar a evolução da web, você deve pensar minimamente em mais alguns assuntos. A seguir, são listados alguns deles para que você possa decidir qual assunto irá pesquisar primeiro.

Internacionalização

Internacionalização é um conceito com aplicação em várias áreas. No contexto da web, está relacionado ao acesso de um site por usuários de qualquer parte do mundo, considerando não só a diferença de idioma, mas também de cultura e costumes. Se o desejo é realizar negócios via web, quanto maior o público, melhor para os negócios. Assim, é necessário considerar o novo público-alvo e criar versões do site para cada cultura diferente. Isso englobará adaptações do idioma, fuso horário, formato de números, sistemas métricos, moedas, aspectos legais e até mesmo nas características do layout, tais como ícones, cores e tamanhos dos elementos, pois eles podem ter um impacto diferente em determinadas culturas. Em Disponível em: <<http://www.w3.org/International/quicktips/Overview.pt.php>>. Acesso em: 19 out. 2015, são listadas algumas dicas para internacionalização de um site.



Vídeo 06 - Hospedagem de Sites

Manipulação de ferramentas de design gráfico

Um dos aspectos que citamos no curso é de que a aparência do site é importante para conquistar o usuário, e, então, falamos sobre a metáfora visual. Você pode criar uma metáfora visual sem nenhuma ferramenta gráfica, mas, se você souber manipular essas ferramentas, melhor, pois elas possibilitam a criação de storyboards e a adaptação ou criação de elementos gráficos que se encaixam adequadamente à metáfora idealizada. Dentre as ferramentas mais utilizadas, estão

o CorelDRAW versão X7 e o Photoshop versão CS5, ambas pagas. Porém, há também ferramentas gratuitas, por exemplo, a ferramenta Pencil, disponível em <<http://pencil.evolus.vn/>>.

O Photoshop CS2 foi liberado para utilização gratuita. Você pode baixa-lo nesse [link](#).

Desenvolvimento para web

Atualmente, os sites já não são mais somente informações para serem consultadas. Eles prestam serviços, muito além de ler os dados de um formulário e guardar esses dados. Um exemplo conhecido é o site de um banco. Nele, você pode acessar sua conta, como se estivesse em um caixa eletrônico e fazer diversos tipos de consultas, transferências, pagamentos e investimentos. Essas páginas não são especificadas com HTML e CSS puros. Em vez disso, linguagens de programação são usadas, tais como javascript, php, asp, ruby on rails, ou java. No curso Desenvolvimento Web do IMD, você aprenderá algumas linguagens de programação e outros assuntos interessantes relacionados a esse tópico.

Projetar para dispositivos móveis

Com o barateamento e avanço dos dispositivos móveis, não só laptops, mas, principalmente, os telefones celulares e tablets, surge a possibilidade de muitos negócios. Assim, é necessário considerar as características desses dispositivos e criar versões de sites para eles. As características mais impactantes são: tamanho da tela, acesso via um teclado minúsculo e tempo de acesso dos usuários a um site por meio desses dispositivos. Em: <<http://www.w3.org/standards/webdesign/mobilweb>>; <<http://www.w3.org/TR/mobile-bp/>>, acesso em: 19 out. 2015, são descritas as recomendações da W3C para autoria de páginas para dispositivos móveis.

Crossbrowser

Você aprendeu (X)HTML e CSS, mas não pense que tudo funciona como você imagina. Diferentes browsers dão suporte a essas tecnologias de maneiras diferenciadas. Por exemplo, um site que você projeta usando o Google Chrome como

teste não necessariamente vai funcionar da mesma forma no Internet Explorer ou demais browsers. E pior, há diferenças significativas também entre as diversas versões de um mesmo browser. Sendo assim, você precisa criar o hábito de testar o site em diferentes browsers e diferentes versões. Em alguns deles, muitos erros acontecerão, principalmente no Internet Explorer 6 e anteriores, e será necessário saber como corrigir esses erros. Afinal de contas, você deseja que todos tenham acesso ao site. Você pode começar conhecendo uma lista dos erros mais frequentes nos diferentes browsers e como corrigi-los em <<http://www.positioniseverything.net/>>, Acesso em: 19 out. 2015, acessando o Browser Shots. Disponível em:<<http://browsershots.org/>>. Acesso em: 19 out. 2015, que é um site que gera uma figura do site escolhido em diferentes navegadores e versões.

Saber o preço de seu trabalho

Se você for trabalhar para uma empresa ou mesmo decidir ter o próprio negócio, é necessário saber quanto custa o seu trabalho. Dessa forma, você poderá estimar qual o custo de um projeto e quanto tempo levará para concluí-lo. Para isso, é necessário saber o que levar em conta e montar uma planilha dos custos do projeto. Em Watrall e Siarto(2009), é ilustrado como esses dados são levantados e como uma planilha pode ser feita.

Enfim, esses são apenas alguns dos assuntos importantes nos quais você pode iniciar suas pesquisas. Com a base que você já tem de HTML e CSS, está apto a iniciar o estudo detalhado de qualquer um deles, ou mesmo de todos. Então, bons estudos!



Vídeo 07 - *Google PageSpeed Insights*

Leitura Complementar

- Planejando a implementação de acessibilidade à Web:
<<http://maujob.com/w3c/wai-impl.html>>. Acesso em: 19 out. 2015.
- Guia de desenvolvimento para dispositivos móveis — em inglês:
<<http://www.pinceladasdawe.com.br/blog/2007/03/21/guia-de-desenvolvimento-web-para-dispositivos-moveis/>>. Acesso em: 19 out. 2015.
- Melhores práticas para internacionalização:
<<http://www.w3.org/TR/2007/NOTE-i18n-html-tech-lang-20070412/>>. Acesso em: 19 out. 2015.
- Site sobre acessibilidade em português:
<<http://www.bengalalegal.com/capitulomaq.php>>. Acesso em: 19 out. 2015.
- Tutorial hacks:
<<http://maujob.com/tutorial/hacks-css.php>>. Acesso em: 19 out. 2015.

Resumo

Nesta aula, você aprendeu o que é acessibilidade e conheceu as quatro recomendações para tornar um site mais acessível. Além disso, você aprendeu também algumas técnicas para tornar um site fácil de manter e conheceu alguns assuntos essenciais a serem aprendidos após este curso.

Autoavaliação

1. Conceitue acessibilidade.
2. Explique o que você entende por cada uma das 4 recomendações da WCAG2.0.
3. Cite 4 técnicas para tornar um documento Web acessível.
4. Use a ferramenta Wave para avaliar a acessibilidade do site que você desenvolveu para um colecionador, cite quais os tipos de erros encontrados.
5. Use o JAWS para ler a página que você desenvolveu para um colecionador, cite os principais problemas encontrados.
6. Use as recomendações descritas nesta aula para melhorar a acessibilidade do site que você desenvolveu para um colecionador.
7. Dentre as atividades desempenhadas por um webdesigner mencionadas nesta aula, qual a que você tem mais afinidade? Faça uma pesquisa sobre essa atividade e cite 3 vantagens e 3 desvantagens/dificuldades relativas à atividade escolhida.
8. Explique, com suas palavras, porque é necessário manter um site sempre atual.
9. Cite duas formas de tornar um site mais fácil de manter.
10. Dentre os assuntos não abordados neste curso, e citados anteriormente, qual o que você gostaria de estudar imediatamente e por quê?

Referências

WATRALL, Ethan; SIARTO, Jeff. **Use a cabeça:** web design. Rio de Janeiro: Alta Books, 2009.