Nós entrevistamos o desenvolvedor Bruno Welber, que além de aluno da plataforma, trabalha com consultoria de acessibilidade. Ele nos ajudou a fazer melhorias na página da Apeperia e deu dicas de como deixar nossos sites mais acessíveis.

**Bruno:** Olá, pessoal da Alura! Sou aluno da plataforma há dois anos, fiz excelentes cursos, e trabalho com desenvolvimento web, mais focado em acessibilidade. Desde 2015, estou fazendo testes nisto, tanto para web quanto para mobile. Trabalho também com back-end.

**Natan:** Só por curiosidade, com quais linguagens você trabalha no back-end?

**Bruno:** PHP, Java, C#...

**Natan:** Basicamente, o que vier você está programando.

**Bruno:** Basicamente. Se aparecer um projeto legal, fazemos.

**Natan:** Legal. Te chamei aqui pois queria que você testasse a acessibilidade do nosso novo projeto, que é a **Apeperia**, uma startup que desenvolve aplicativos para pequenas e médias empresas. Conforme você for navegando, gostaria que fosse me informando sobre o que está pensando, sentindo, o que você esperava, e coisas assim.

**Bruno:** Estamos no site...

**Natan:** Essa voz aí, é uma entidade, algo assim?

**Bruno:** Essa voz do astral que está conversando comigo agora é o leitor de telas **NVDA**, gratuito e open source, possível de ser baixado no [site oficial](https://www.nvaccess.org/) para ser usado tranquilamente. Esta voz específica que estamos utilizando não é próprio do NVDA, e sim do Windows, sendo disponível apenas para a versão 10 do Windows - a partir da Creator -, é legal ressaltar. Uso a voz do "Daniel" porque a do NVDA é um tanto "robotizada".

**Natan:** Mostrarei o NVDA com calma nos próximos vídeos, não se preocupem com isso.

**Bruno:** [leitor de tela lê em voz alta conforme Bruno navega a página pressionando a tecla "TAB" do teclado] Acredito que isto seja um cabeçalho... Acho que estou no Brasil, então não faz sentido o site alternar entre inglês e português. Aqui já existe uma falha que atrapalha bastante a navegação.

**Natan:** Legal, veremos como corrigir isso.

**Bruno:** Terei que desativar essa detecção de idioma que, se usada corretamente, é útil. Para isso, desabilitaremos o box de "Alternância automática de idioma quando suportado" na janela "Opções de voz".

**Natan:** Então como o NVDA está em português, estamos desabilitando essa alternância de idiomas.

**Bruno:** Exatamente.

**Natan:** Faz sentido. Mas isso é um "jeitinho" que você encontrou para lidar com esse problema de acessibilidade, certo?

**Bruno:** Sim, pois o correto seria a leitura do texto ser feita de acordo com o idioma local. Da maneira como estava, era estranho. [Bruno continua navegando usando o leitor de tela] Botão "Conheça os planos", vamos ver se ele faz algo. Faz?

**Natan:** Não aconteceu nada.

**Bruno:** Então isso não é um botão. [risos]

**Natan:** É só uma pegadinha.

**Bruno:** Já que não aconteceu nada, vou continuar pela página. "Cabeçalho nível 1 tempo", tudo bem, mas o tema principal da página é a criação de aplicativos ou é gestão de tempo?

**Natan:** Boa pergunta, seria o tempo de São Paulo, algo assim.

[leitor de tela lê "Cabeçalho nível 1 foco"]

**Bruno:** Mais um cabeçalho... O "Sobre" está sobrando literalmente.

**Natan:** Aí aparece um texto gigantesco.

**Bruno:** "Ícone do Android gráfico", "Ícone da Blackberry gráfico"... é meio redundante, já sei que é um gráfico, não importa o que seja. "Desenvolvemos aplicativos para todas as plataformas" já é compreensível pelos gráficos, essa frase deveria estar antes deles. Este "assinar plano" se refere ao plano de cima ou de baixo?

**Natan:** Legal, é interessante reparar que visualmente tudo parece estar claro para nós, então são preocupações que não costumamos ter mesmo, mas aqui no curso iremos pensar um pouco nessas questões.

**Bruno:** Até mesmo por conta do requisito de você não apresentar uma informação somente por cores como está sendo feito neste caso...

**Natan:** Exatamente. Isso foi uma das coisas que abordamos no curso anterior, então, se você está começando por este, recomendo fortemente que comece por [Acessibilidade web: introdução a designs inclusivos](https://www.alura.com.br/curso-online-acessibilidade-web-design-inclusivos), que é mais voltado para design, só que é interessante para não cometer gafes deste tipo.

**Bruno:** "Notícia 1, notícia 2, notícia 3".

**Natan:** Bem descritivo, não?

**Bruno:** Totalmente, se forem trocadas amanhã, serão "Notícia 4, notícia 5 e notícia 6"? "Receber destaques por e-mail": vamos tentar clicar nesse botão, que para mim é outro que não faz nada.

**Natan:** Fez, sim! Abriu um modal, veja que bacana.

**Bruno:** Te juro que não estou vendo nada. [risos]

**Natan:** Que sacanagem. Tem que bater no frontender que fez este site.

**Bruno:** Concordo plenamente. Continuando: "Assine nossa newsletter para receber notícias técnicas sem spam!", creio que isto faça parte do botão, pelo jeito.

**Natan:** Isto é como se fosse uma descrição, um parágrafo embaixo de um título.

**Bruno:** Automaticamente deduziria que para fechar este modal teria que usar a tecla "esc", porém, nada acontece.

**Natan:** Isso aí faz parte dos problemas que iremos corrigir durante o curso, mais especificamente na parte de Javascript. Bruno, se você fosse fechar a janela, como você faria?

**Bruno:** Teria que procurar o "X" [ícone de fechar], vamos ver se pelo menos isso o desenvolvedor deixou apresentável. "Botão xX", será que funciona?

**Natan:** Funcionou! Qual seria o resumo, o que você achou da acessibilidade deste site?

**Bruno:** Acho que realmente teria que dar um castigo para quem fez isso, como fazê-lo ficar uma semana usando o NVDA.

**Natan:** Bruno, obrigado por ter vindo, e topado participar deste vídeo do começo do curso. Vamos melhorar a acessibilidade dos nossos projetos, porque não adianta nada deixar o site esteticamente impecável, porém, inacessível.

**Bruno:** Valeu pessoal.

Como vimos com o Bruno, não temos alguém nos indicando 24 horas onde o cursor do mouse está localizado na tela. O Bruno usa um software para isso, os chamados **leitores de tela**, que funcionam indicando e lendo a localização do foco. Dentre as opções disponíveis, estão o [JAWS](http://www.freedomscientific.com/Products/Blindness/JAWS), o [NVDA](https://www.nvaccess.org/) e, no caso do Mac, o [VoiceOver](https://www.apple.com/br/accessibility/mac/vision/" \t "_blank) - leitor que vem integrado ao sistema operacional. Há também o **Narrador** no Windows, e outros.

Qual deles é o mais interessante? Nós precisamos fazer testes e ver o que de fato o usuário usa. Se ele utilizar o Windows 7, por exemplo, teremos que encontrar a melhor solução, testando as aplicações com este navegador. Da mesma forma, para saber qual leitor de tela é o melhor ou mais utilizado, pode-se consultar o site [WebAIM](https://webaim.org/" \t "_blank), totalmente focado em acessibilidade web.

Foi feita uma pesquisa em 2017, que indica qual leitor de tela é o mais utilizado, e o JAWS, junto com o NVDA, têm juntos 78.5% do *market share*, enquanto o VoiceOver é utilizado por 11.7%, pois é bem grande a quantidade de usuários de Mac. O JAWS, apesar de ser o mais usado, tem decaído em uso conforme a passagem dos anos.

A barreira do JAWS é que a licença para a sua instalação em português custa cerca de R$8 mil, sendo um investimento alto para nós desenvolvedores testarmos em nossas interfaces. No caso do VoiceOver - disponível no Mac -, não há custo adicional, assim como o Narrador do Windows. No nosso projeto, usaremos o NVDA, que aceita doações via PayPal, porém, não é compatível com Mac.

Na mesma pesquisa, cerca de 73% dos usuários de leitores de tela usam o Windows como sistema operacional, e por este motivo focaremos mais nele. Veremos também que o navegador mais utilizado por quem possui leitores de tela é o Firefox (41%), seguido pelo IE11 (23.3%) e pelo Chrome (15.5%). O Bruno, no entanto, afirmou que o Chrome não é tão interessante em termos de compatibilidade com leitores de tela.

Instalaremos o NVDA, que começa falar desde o início. Para evitarmos isto, manteremos a tecla "Ctrl" pressionada, enquanto seguimos a instalação definindo as configurações conforme o que precisamos. Na tela de boas-vindas, vamos manter marcado o *checkbox*de "Usar CapsLock como uma tecla modificadora do NVDA".

Para que a voz não atrapalhe a aula por enquanto, vamos desativá-la clicando no ícone do programa no canto superior direito da tela, e em "Preferências > Opções de voz..." e diminuindo todo o volume.

Provavelmente, você deve ter notado que a voz do leitor é "robotizada", ou seja, bastante sintética e artificial. No Windows, há duas opções:

* "Maria";
* "Daniel".

Usaremos essas opções no NVDA, faremos isto, clicando com o lado direito do mouse sobre o ícone do NVDA mais uma vez, e depois em "Preferências > Sintetizador...". Na janela que será aberta, trocaremos eSpeak NG por Vozes Windows OneCore em "Sintetizador:". Ao clicarmos em "OK", passaremos a ouvir a voz da Maria, mas você tem a liberdade para selecionar o "Daniel". É possível aumentar ou diminuir o volume, basta fazer o ajuste.

Na mesma janela de configurações de opções de voz, manteremos o *checkbox* de "Alternância automática de idioma quando suportado", e em "Grau de sinais de pontuação:", poderemos deixar muito, o que significa que será dada muita ênfase nas pontuações. As demais definições poderão ser mantidas de acordo com o padrão.

Se você for testar algo via teclado durante o curso, o NVDA lê cada caractere digitado - isto pode ser desativado em "Preferências > Opções de teclado...". Outro ponto interessante é o **Exibidor de Fala do NVDA** existente em "Ferramentas", que abre uma janela mostrando tudo o que está sendo feito conforme é passado o mouse ou alterado o foco. Isto é útil para quando não queremos ficar ouvindo o leitor de telas o tempo todo, pois temos também a opção de ler.

É possível salvar estas configurações ao clicarmos com o lado direito do mouse no ícone do NVDA e em "Salvar configuração", bem como criar perfil de configuração e levar para outro desktop ou notebook, por exemplo.

**Por que o NVDA?**

Focaremos no uso do **Windows + Firefox + NVDA** neste curso, que é um dos combos mais utilizados segundo a [pesquisa da WebAIM sobre o uso de leitores de tela](https://webaim.org/projects/screenreadersurvey7/) mostrada em aula.

**Windows**

Acesse o [site do NVDA para efetuar seu download](https://www.nvaccess.org/) .

Instale-o e configure-o conforme mostrado em aula.

E se o seu estiver em inglês, pode baixar os pacotes de idiomas seguindo o [tutorial da própria Microsoft](https://support.microsoft.com/pt-br/help/22797).

**Mac e outros sistemas**

O NVDA atualmente não está disponível para esses sistemas operacionais.

Você pode acompanhar o conteúdo do curso sem usar um leitor de tela, mas recomendo tentar usar o [Voiceover no seu Mac](https://www.apple.com/br/accessibility/mac/vision/" \t "_blank) ou o [Orca no seu Linux.](https://help.gnome.org/users/orca/stable/index.html.pt_BR)

Ou mesmo procurar algum outro nessa [lista de leitores de tela da Wikipedia](https://en.wikipedia.org/wiki/List_of_screen_readers).

[Arquivos iniciais usados na aula (ZIP, 16MB)](https://github.com/designernatan/curso-acessibilidade-web-front-end-1/archive/14e5c4ffd5f7f6d767c8a88e88659f5cec5eb253.zip).

É importante destacar que se o seu Windows 10 estiver em inglês, provavelmente, as vozes disponíveis de narrador virão apenas em inglês, por isso, temos disponível o [que contém um pequeno tutorial](https://support.microsoft.com/pt-br/help/22797/windows-10-narrator-tts-voices) de como instalar as vozes no idioma português. Você também pode instalar outros idiomas e usá-los nos seus testes.

No exibidor de fala do NVDA (*Screen Reader*) a fonte é pequena, se for do seu interesse ampliá-la, basta pressionar o atalho "Ctrl + mouse scroll"

Neste projeto, usaremos o editor **Brackets**, pois ele possui a opção "Live Preview", localizado na parte superior direita da tela e simbolizado pelo ícone de um raio.

Ao selecionarmos esse ícone, automaticamente é criada uma *local server*, e se quisermos modificar algum elemento, por exemplo, o título do projeto, o elemento será selecionado dentro do editor quando clicarmos no *browser*. Quaisquer alterações feitas são replicadas imediatamente no local server, então não precisamos ficar salvando o documento.

Outro recurso interessante do Brackets: ao selecionarmos uma classe no editor e pressionarmos o atalho "Ctrl + E", surgirá uma caixa que nos permitirá alterar o código CSS.

A seguir, selecionaremos a classe .secaoDestaque-titulo do projeto. Se fizermos alguma alteração, ela será replicada automaticamente no nosso código. Mas como estamos trabalhando com o arquivo de estilo, devemos salvar as alterações.

Você tem a possibilidade de usar outro editor de sua preferência, com Visual Studio Code, Sublime. Eu vou usar o Brackets, por considerar que ele facilita testar nosso projeto.

, com algumas extensões instaladas. Para visualizá-las, selecionaremos o ícone representado por uma peça de lego, localizado na parte superior direita da tela.

Teremos acesso à uma caixa de diálogo. Na parte superior esquerda, selecionaremos a opção "Availble" e no campo de busca podemos pesquisar pelas extensões desejadas.

Instalei a extensão do tema "Monokai", que deixará o editor semelhante ao Sublime, e também baixei "Indentator", com o qual podemos indentar o código automaticamente. Recomendo ainda a instalação do "Emmet", bastante útil para agilizar a escritura do código.

No curso, usaremos o navegador Firefox, por ser o mais utilizado pela maioria dos usuários de NVDA (juntamente, com IE). O site do projeto "Apeperia" está responsivo, vemos que o conteúdo se ajusta ao diminuirmos o tamanho da janela. No entanto, ainda tem alguns pontos que precisam de ajustes. Vamos analisar o código da página.

Observe que os componentes CSS foram separados em vários arquivos. Por exemplo, todo conteúdo relacionado a cabecalho, se encontrará armazenado no arquivo correspondente, chamado cabecalho.css. Todos os arquivos foram chamados no index.html, como veremos abaixo:

!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Apeperia - aplicativos sob medida</title>

<link href="https://fonts.googleapis.com/css?family=Montserrat:300,400,700|Open+Sans:300,400,700" rel="stylesheet">

<link rel="stylesheet" href="css/reset.css">

<link rel="stylesheet" href="css/tema-padrao.css">

<link rel="stylesheet" href="css/secaoDestaque.css">

<link rel="stylesheet" href="css/cabecalho.css">

<link rel="stylesheet" href="css/diferenciais.css">

<link rel="stylesheet" href="css/secaoSobre.css">

<!-- ... -->

Para acessarmos rapidamente um arquivo no Brackets, podemos utilizar o atalho "Ctrl + Shift + O". Com isso, será aberta uma janela na qual pesquisaremos pelo nome do arquivo desejado e pressionaremos "Enter". Então, o arquivo será aberto.

E para relembrar, com o atalho "Control + E", será aberta uma caixa na qual poderemos fazer alterações em um trecho específico.

Recomendo novamente, que você tenha um conhecimento prévio em HTML e CSS.

O próximo passo será conhecermos a estrutura básica do site. Temos o cabecalho container, na tag <header>. Temos o conteúdo principal que é o elemento <main> e temos <footer>. Sobre o carrossel, nos aprofundaremos mais adiante no curso.

<header class="cabecalho container"> </header>

<main class="conteudo"> ..</main>

<footer class="rodape"> </footer>

Estamos invocando uma série de arquivos CSS, como já foi dito. Estamos, também, chamando o arquivo da fonte Montesserat do Google e a fonteOpen+Sans.

!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Apeperia - aplicativos sob medida</title>

<link href="https://fonts.googleapis.com/css?family=Montserrat:300,400,700|Open+Sans:300,400,700" rel="stylesheet">

Outros pontos que podemos comentar brevemente sobre o projeto: o logo do site Apeperia está em svg. Temos um menu de navegação, em um código bastante **semântico**, ou seja, ele usa os elementos da linguagem de marcação em conformidade com o seu significado. E não simplesmente, adicionar a tag <nav>.

Na tag <main>, em cada section, teremos um arquivo CSS a parte, a não ser que estejamos tratando de um componente específico.

Apresentamos o arquivo HTML, sugiro que você se aventure em observar as propriedades. Analise se as declarações e propriedade estão claras para você.

A seguir, continuaremos melhorando a acessibilidade do nosso projeto.

Vamos começar a realizar modificações produtivas em nosso projeto. Veremos alguns atalhos interessantes do NVDA. Se pressionamos a tecla "K", iremos navegar entre todos os links disponíveis na página. Caso você queira voltar para o link anterior, basta pressionar "Shift + K".

Mas como os deficientes visuais possuem uma noção geral sobre do que se trata um determinado site? Normalmente os destaques estão na tag <h1>, <h2> e assim por diante, ou seja, o usuário navega entre os títulos da página utilizando a tecla "H". Ao pressionar a tecla "H", o leito de tela narra "Aplicativos Sobre Medida", o cabeçalho do nível 1.

É interessante desabilitarmos a leitura do mouse, caso o contrário, toda a vez que o cursor passar por um link a informação será registrada na read screen do NVDA. Para realizar essa desabilitação,selecionaremos as configurações no NVDA, e a opção "preferências>opções de mouse>" e desabilitamos a opção "Habilitar rastreamento do mouse".

Os títulos são diferenciados por nível 1 ou 2, porque escrevemos essa informação no front-end por meio de <h1> e <h2>. Se pressionarmos continuamente o atalho "H" seremos conduzidos para os headings (ou títulos) da página, que são "Tempo", "Foco", "Especialistas", "Sobre", "Planos" e assim por diante.

Para sabermos do que se trata nosso site de forma geral, leremos o título "Aplicativos na medida" e em seguida, encontraremos um pequeno texto. No entanto, no canto superior esquerdo, há a presença de um link gráfico com o logo "Apeperia" que se apresenta desnecessário na ótica de um site inclusivo, bastaria um link escrito "home" que direcionasse o usuário para a parte principal da página.



Outra questão que devemos os atentar: alguns leitores de tela não diferenciam o <h1>da svg, do <h1> da classe secaoDestaque-titulo,por exemplo. Compare os dois trechos de código:

<h1>

<svg width="205" height="59" viewBox="0 0 205 59" version="1.1" xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink">

<desc>Created using Figma</desc>

Código da classe secaoDestaque-titulo:

<h1 class="secaoDestaque-titulo">

APLICATIVOS NA MEDIDA

</h1>

O mais importante é o título "Aplicativos na medida", portanto, deixaremos apenas esse texto com <h1> e retiraremos de outras porções do código, como esta que corresponde ao cabeçalho da página:

<header class="cabecalho container">

<a href="#" class="cabecalho-logo">

<h1>

<svg with="205" height="59" viewBox="0 0 205 59"

version"1.1" xmlns-"http://www.w3.org/2000/svg"

xmlns:xlink="http://www.w3.org./1999/xlink">

</svg>

</h1>

Normalmente os profissionais CEO e otimização para mecanismos de busca sugerem um uso estratégico das <h1>, mas essa questão é diferente quando estamos pensando em acessibilidade.

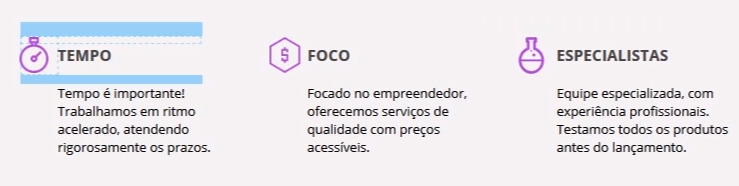
O pequeno texto que segue o título também é muito importante para ser simplesmente um parágrafo,afinal, fornece o contexto geral do site. Por isso, o envolveremos em <h2>para que assuma o caráter de subtítulo, apesar de não sê-lo exatamente.

<h2 class="secaoDestaque-texto">

A Apeperia é uma startup com um jeito inovador de comprar e montar aplicativos mobile e web para pequenas e médias empresas

</h2>

Observem as seguintes listas:



Como queremos manter apenas o título principal da página com <h1>, retiraremos as tags <h1> do título das listas de "tempo", "foco" e "especialistas" e substituiremos por <h2>.

<h2 class="diferenciais-diferencial-titulo diferenciais-diferencial-titulo--tempo">

Tempo

</h2>

Essa alteração não modificará o tamanho das fontes na página. Faremos a mesma modificação no texto SOBRE, o envolvendo em <h2> ao invés de <h1>.

<h2 class="secaoSobre-titulo tituloPadrao">

SOBRE

</h2>

Na área de "Planos", percebam que existe uma hierarquia de informações: "Planos" anuncia a sessão, e logo abaixo temos as modalidades disponíveis que são "Start", "Ultra" e "Mega".



Não faz sentido que o título "Planos" esteja no mesmo nível de "Start" ou "Ultra", por isso, envolveremos o título dessas modalidades de plano na tag <h3>. Em seguida, envolveremos os preços de cada modalidade na tag <h4>.

<div class"secaoPlanos-plano secaoPlanos-plano--

ultra">

<h3 class="secaoPlanos-plano-titulo">Ultra</h3>

<h4 class="secaoPlanos-plano-preco">

R$ 1500

</h4>

Prosseguiremos realizando uma varredura no código procurando <h1>s para que não haja complicações com os leitores de tela.

Modificaremos DESTAQUES o envolvendo em <h2>

<section class="secaoDestaques">

<div class="container">

<h2 class="secaoDestaques-titulo

tituloPadrao">DESTAQUES</h2>

Faremos o mesmo procedimento com News sem spam, o envolvendo em <h2>. O texto Assine nossa newsletter para receber notícias técnicas sem spam! envolveremos em <h3>.

<h2 class="dialogoNewsletter-titulo">News sem spam</h2>

h3 class="dialogoNewsletter-texto">

Assine nossa newsletter para receber notícias técnicas sem spam!

</h3>

Seguiremos no caso do texto CONTATO.

<section class="secaoContato">

<div class="container">

<h2 class="tituloPadrao">

CONTATO

</h2>

Por último, o texto INSTITUCIONAL

<section class="secaoInstitucional">

<div class="container">

<h2 class="tituloPadrao">

INSTITUCIONAL

</h2>

Ao atualizarmos a página veremos que, visualmente, nada se modificou. Quando habilitamos o som do NVDA e utilizamos o atalho "H" percebemos que a apresentação do site no sentido da acessibilidade melhorou bastante, e o usuário consegue navegar entre os níveis diferentes de informação.

Não existe uma regra estática para projetar páginas acessíveis, precisamos sempre analisar cada contexto para saber quais as modificações devem ser realizadas.

Atualmente é aceitável, semanticamente, uma página de um site possuir diversos <h1>distribuídos em seções independentes.

Contudo, isso pode prejudicar a acessibilidade do seu site pois:

Alguns leitores de tela podem se perder.

Apesar do NVDA não se perder e ditar o nível do *heading* em foco, outros leitores de tela (principalmente versões antigas) podem ler apenas o primeiro <h1> ou simplesmente se perderem, e por consequência prejudicar a experiência do usuário no site. Principalmente usando o atalho para ler o que seria o ponto mais importante do site (comumente H + 1).

Neste primeira aula, introduzimos ferramentas e fluxos de como começar a solucionar nossos problemas de acessibilidade e aprendemos bastante:

**Uso do leitor de tela:**

* Que pessoas cegas ou com baixa visão podem contar com o auxílio de um software que lê o que está na tela, os chamados leitores de tela.

**Usando o NVDA:**

* Vimos, configuramos e aprendemos atalhos do NVDA, leitor de tela escolhido para testar o projeto Apeperia, a fim de deixar seu uso em testes menos maçante para quem não está acostumado com a ferramenta. E também habilitamos seu "exibidor de tela" para não dependermos de ouvir sua leitura.

**Semântica no HTML:**

* Tomar cuidado com a marcação de conteúdo quando estamos utilizando elementos HTML5. Vale mais por uma <div> do que usar um <aside> ou <details>erroneamente.

**Testes, testes, testes:**

* O melhor caminho para deixar nosso projeto inclusivo não é apenas seguir regras e *guidelines*, mas testá-las por conta para começarmos a compreender os caminhos e dificuldades que muitas pessoas podem ter ao usar nosso produto e/ou serviço. Empatia.

Faça as modificações feitas em aula a fim de deixar o projeto inicial mais acessível tendo como foco o publico cego e de baixa visão.

Lembre-se de pontos como:

* evitar mais de um <H1> na página;
* usar HTML5 semântico que faça sentido;
* configurar o NVDA conforme visto em aula.

[Acessibilidade - por que deixarmos de ser amadores para um público que espera mais de nós (SlideShare)](https://pt.slideshare.net/MarinaDomingues7/acessibilidade-por-que-deixarmos-de-ser-amadores-para-um-pblico-que-espera-mais-de-ns)

Palestra da [Marina Domingues.](https://www.linkedin.com/in/marina-domingues/)

Espero que você tenha se aventurado no código html que foi disponibilizado. É importante essa observação para que se entenda a estrutura do projeto.

Nesta aula, iremos nos atentar a alguns detalhes importantes. Analisaremos como se dá a navegação do NVDA ao longo da página. Percebam que o leitor de tela lê os primeiros itens "Sobre", "Planos", "Blog", "Institucional" e "Contato" com um forte sotaque norte-americano.



Ao observarmos nosso código, veremos o atributo que define o idioma que será interpretado pelo leitor de tela e até mesmo pelo Google, está como en.

<!DOCTYPE html>

<html lang="en">

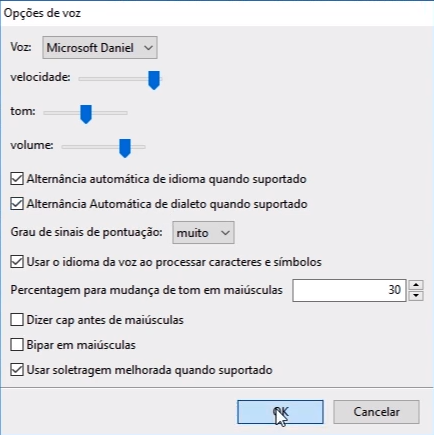
<head>

<meta charset="UTF-8">

<title>Apeperia - aplicativos sob medida</title>

Alteraremos esse atributo para pt-br. Com isso, a leitura dos itens da página está no idioma que nos interessa, facilitando a compreensão do conteúdo para o usuário deficiente visual.

Essa alteração no idioma lido por parte no NVDA só está ocorrendo porque as opções "Alternância Automática de dialeto quando suportado" e "Alternância de idioma" estão habilitadas. Para configurar essas opções, basta selecionar "Preferências>Opções de Voz" no NVDA.



Mas no caso de uma palavra em inglês? Para testarmos isso, alteraremos o texto "APLICATIVOS NA MEDIDA" para "The book is on the table".

<section class="secaoDestaque">

<h1 class="secaoDestaque-titulo">

The book is on the table

</h1>

Neste caso, o leitor de tela lê uma frase em inglês com um sotaque brasileiro. Isso se deve ao fato de termos configurado que todo html seria em pt-br. Se precisamos que um fragmento de código seja lido em um idioma específico, teremos de inserir o atributo leng e acionar a opção en.

<section class="secaoDestaque">

<h1 class="secaoDestaque-titulo" lang="en">

The book is on the table

</h1>

Isso vale para outros idiomas, como espanhol, por exemplo. Neste caso, adicionamos o atributo es.

<section class="secaoDestaque">

<h1 class="secaoDestaque-titulo" lang="es">

Si, pero no mucho

</h1>

É importante prestarmos atenção ao lang, pois ele interfere diretamente na experiência do usuário na página. Se você quiser saber mais sobre isso, há disponível uma palestra nos exercícios que analisa com mais profundidade essa questão.

Você foi contratado como consultor de acessibilidade pela [Alura Língua](https://www.aluralingua.com.br/" \t "_blank), nossa plataforma de cursos online de inglês.

A seção de diferenciais da plataforma deve ficar da seguinte forma:



E o código HTML está da seguinte forma:

<ul class="vantagens-lista">

<li class="vantagens-lista-item">

<img src="/imagens/icone-spaced.png">

<h1 class="vantagens-item-titulo">

Spaced Repetitions

</h1>

<p class="vantagens-item-texto">

A metodologia se adapta ao seu conhecimento. Você evolui naturalmente nos exercícios, até fixar todo o conteúdo.

</p>

</li>

<li class="vantagens-lista-item">

<img src="/imagens/icone-flashcards.png">

<h1 class="vantagens-item-titulo">

FlashCards

</h1>

<p class="vantagens-item-texto">

Exercícios objetivos e focados em uma habilidade específica, para praticar tudo que aprendeu em cada lição.

</p>

</li>

<li class="vantagens-lista-item">

<img src="/imagens/icone-speak.png">

<h1 class="vantagens-item-titulo">

Fale Bastante

</h1>

<p class="vantagens-item-texto">

Você é encorajado a repetir e criar frases inteiras, evitando ter de pensar na tradução palavra a palavra.

</p>

</li>

</ul>

Quais principais problemas de acessibilidade esse código pode causar?

Uso de vários <h1> na mesma página.

Pensando em acessibilidade, recomenda-se o uso de um único <h1> por página.

Imagens ilustrativas sendo marcadas como conteúdo.

Importante sempre separar as responsabilidades entre o que é visual e o que é conteúdo. Iconografia **normalmente** tem caráter visual, e o que for visual temos que fazer no CSS. Seja usando um background-image ou um ::after/::before. Você pode ver o código-fonte direto na [home da Alura Língua](https://www.aluralingua.com.br/) .

Daremos prosseguimento na análise da experiência do usuário em um primeiro contato com a página.

Percebam que pela navegação sem mouse, não conseguimos acessar o logo da Apeperia. Por isso, habilitaremos o mouse em "Preferências> Opões de mouse>" e selecionaremos a opção "Habilitar rastreamento mouse".



No momento em que posicionamos o mouse sobre o logo, será exibida a seguinte informação no tela de fala do NVDA: "Created using Figma", que traduzido para o português significa "criado usando Figma".

Ao observarmos a descrição desse elemento no código, notaremos o texto que é lido pelo NVDA. Provavelmente, ao exportarmos os elementos do Figma, esta será uma descrição embutida.

<header class="cabecalho container">

<a href="#" class="cabecalho-logo">

<h1>

<svg width="205" height="59" viewBox="0 0 205 59" version="1.1" xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink">

<desc>Created using Figma</desc>

Como essa descrição é irrelevante, vamos alterá-la para Logotipo da Apeperia.

<header class="cabecalho container">

<a href="#" class="cabecalho-logo">

<h1>

<svg width="205" height="59" viewBox="0 0 205 59" version="1.1" xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink">

<desc>Logotipo da apeperia</desc>

<g id="Canvas" transform="translate(-2230 1001)">

<g id="Logotipo da Apeperia">

No entanto, o elemento <desc> do svg é utilizando exatamente para descrição de algum conteúdo e não para o título da imagem, portanto não se trata de um alt.

Não possuímos um elemento alt, mas podemos utilizar <title> para nomear a imagem e mantermos a descrição original.

<header class="cabecalho container">

<a href="#" class="cabecalho-logo">

<h1>

<svg width="205" height="59" viewBox="0 0 205 59" version="1.1" xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink">

<title>Logotipo da apeperia</title>

<desc>Created using Figma</desc>

<g id="Canvas" transform="translate(-2230 1001)">

<g id="Logotipo do Apeperia">

Com isso, o NVDA lerá o título "Logotipo da Apeperia" e a sua descrição "Created using Figma". Neste caso, a leitura dos dois campos não faz sentido, como no caso de um infográfico ou algo do gênero, portanto deixaremos esse trecho do código comentado. Para fazer isso basta posicionar o mouse na respectiva linha de código e acionar o atalho "Ctrl + ;"

<!-- <desc>Created using Figma</desc>-->

<g id="Canvas" transform="translate(-2230 1001)">

<g id="Logotipo do Apeperia">

Normalmente um deficiente visual não utiliza mouse, então como se dará o acesso ao logo da Apeperia na página? Desabilitaremos a navegação dos títulos acessada pelo atalho "H", que serviria para o usuário adquiria uma noção geral do site e seu tema.

Habilitaremos "Shift + K" para a navegação normal via teclado. Quando o logo da Apeperia for selecionado, o leitor de NVDA exibe o texto "linque". O que ocorre é que <title> deveria ser lido como a descrição do link.

Por exemplo, criaremos um link com o texto "Vai para algum lugar".

<a href="#">Vai para algum lugar</a>

Se substituirmos o texto Vai para algum lugar por uma imagem qualquer, e o adicionarmos no alt, o NVDA lerá a frase "Vai para algum lugar".

<a href="#"><img src="img/10-aplicativos-mais baratos.png" alt= "Vai para algum lugar></a>

Essa estratégia não funcionará com svg, pelo menos não com a title ou desc. Temos duas possibilidades para resolver a questão: ou trocamos o tipo de arquivo, ou inserirmos o atributo title e especificaremos um texto.

<header class="cabecalho container">

<a href="#" class="cabecalho-logo" title="Logotipo da Apeperia">

<h1>

O leitor NVDA registrará "Logotipo da Apeperia linque". No entanto, concluímos que não é interessante inserir a tag <alt> e <title> na imagem. No caso de um arquivo svgdentro de um link, você terá que recorrer ao title>.

Em um determinado ponto do site, temos uma imagem de duas pessoas escrevendo em um quadro branco. No HTML, essa imagem possui já possui no alt o texto Duas pessoas escrevendo em um quadro branco.

<section class="secaoSobre clearfix">

<div class="container">

<img src="img/sobre-apeperia.png" class="secaoSobre-imgDestaque" alt="Duas pessoas escrevendo em um quadro branco">

<h2 class="secaoSobre-titulo tituloPadrao">

Se inserirmos além disso o texto Cursos online de tecnologia dentro title, ao navegarmos pelas imagens do site utilizando o atalho "G", o NVDA registrará o seguinte texto sobre a imagem referenciada Duas pessoas escrevendo em um quadro branco Alura cursos online. Note que a informação não ficou bem construída.

Muitas vezes o CEO de sua empresa solicita que se insira alt e titlee keywords para que o site tenha mais visitações por meio das pesquisa do Google. Mas é muito interessante pensar primeiramente nos usuários e na sua experiência de navegação no site.

Quando deixamos nosso código semântico, acessível e mobile friendly, o Google eleva o site em seu ranking, portanto pensar no usuário é uma ótima estratégia.

Seguiremos trabalhando na tag <alt> da imagem de dois homens escrevendo em um quadro branco, localizada no site da Apeperia.

Analisaremos o texto e se de fato ele é necessário:

<section class="secaoSobre clearfix">

<div class="container">

<img src="img/sobre-apeperia.png" class="secaoSobre-imgDestaque" alt="Duas pessoas escrevendo em um quadro branco">

Essa imagem não passa um conteúdo informativo importante, na verdade, ela é meramente ilustrativa na seção "SOBRE". Por ser ilustrativa, trata-se de estilo, e por isso deveria ser incluída no CSS. Mas ouvimos o CEO e o designer da Apeperia alegando que é importante incluir imagens nesta seção, pois dão harmonia o layout do site - embora não possuam qualquer relevância para quem utiliza um leitor de tela.

Temos duas alternativas:

Podemos melhorar a descrição de alt da imagem, tornando-a mais precisa e inteligente de acordo com os objetivos da Apeperia.

Em vez de "duas pessoas escrevendo em um quadro branco", escreveremos "Dois homens concentrados idealizando um projeto novo em frente a um quadro branco". Percebam como a nova descrição está mais afetiva e precisa, portanto, fique atento no momento da escrita do alt. A descrição deve fazer sentido no contexto de forma perspicaz, ainda que concisa.

Existe um atributo denominado longdesc, que permite que façamos descrições mais longas, mas normalmente 140-160 é uma quantidade suficiente de caracteres, pois o NVDA possui um limite de processamento de leitura. Não podemos escrever descrições muito longas.

A outra opção é simplesmente deixar o alt vazio. Há diversas maneiras de deixar o campo vazio, como alt="" ou alt, isso varia de acordo com o tipo de leitor de tela sendo utilizado pelo usuário.

Deixaremos o alt vazio para observar como o leitor de dela se comporta durante a navegação, e perceberemos que a imagem não é reconhecida pelo leitor.

Precisamos sempre avaliar se a imagem faz sentido quando estamos pensando na estrutura inclusiva de um site, se ela é útil para explicar algo - por exemplo, um infográfico necessita de uma descrição precisa. Mas se ela apenas servir para fins ilustrativos, talvez, seja dispensável.

Outro exemplo interessante é o *ecommerce*, em que é necessário que haja uma imagem do produto. No caso, é interessante que a descrição da imagem seja bastante precisa, especificando características como cor, tamanho ou voltagem.

Seguiremos trabalhando na tag <alt> da imagem de dois homens escrevendo em um quadro branco, localizada no site da Apeperia.



Analisaremos o texto e se de fato ele é necessário:

<section class="secaoSobre clearfix">

<div class="container">

<img src="img/sobre-apeperia.png" class="secaoSobre-imgDestaque" alt="Duas pessoas escrevendo em um quadro branco">

Essa imagem não passa um conteúdo informativo importante, na verdade, ela é meramente ilustrativa na seção "SOBRE". Por ser ilustrativa, trata-se de estilo, e por isso deveria ser incluída no CSS. Mas ouvimos o CEO e o designer da Apeperia alegando que é importante incluir imagens nesta seção, pois dão harmonia o layout do site - embora não possuam qualquer relevância para quem utiliza um leitor de tela.

Temos duas alternativas:

Podemos melhorar a descrição de alt da imagem, tornando-a mais precisa e inteligente de acordo com os objetivos da Apeperia.

Em vez de "duas pessoas escrevendo em um quadro branco", escreveremos "Dois homens concentrados idealizando um projeto novo em frente a um quadro branco". Percebam como a nova descrição está mais afetiva e precisa, portanto, fique atento no momento da escrita do alt. A descrição deve fazer sentido no contexto de forma perspicaz, ainda que concisa.

Existe um atributo denominado longdesc, que permite que façamos descrições mais longas, mas normalmente 140-160 é uma quantidade suficiente de caracteres, pois o NVDA possui um limite de processamento de leitura. Não podemos escrever descrições muito longas.

A outra opção é simplesmente deixar o alt vazio. Há diversas maneiras de deixar o campo vazio, como alt="" ou alt, isso varia de acordo com o tipo de leitor de tela sendo utilizado pelo usuário.

Deixaremos o alt vazio para observar como o leitor de dela se comporta durante a navegação, e perceberemos que a imagem não é reconhecida pelo leitor.

Precisamos sempre avaliar se a imagem faz sentido quando estamos pensando na estrutura inclusiva de um site, se ela é útil para explicar algo - por exemplo, um infográfico necessita de uma descrição precisa. Mas se ela apenas servir para fins ilustrativos, talvez, seja dispensável.

Outro exemplo interessante é o *ecommerce*, em que é necessário que haja uma imagem do produto. No caso, é interessante que a descrição da imagem seja bastante precisa, especificando características como cor, tamanho ou voltagem.

Atalhos

H – seremos conduzidos para os headings (ou títulos) da página

G – Navegar entre as imagens do site

K – navegar entre os links das pagina

Shift + K – Voltar no link anterior