

2 SISTEMA DE BOLETINS INFORMATIVOS

Durante a disciplina de criação de conteúdo foi proposto um trabalho para a criação de uma página para mostrar os boletins informativos. Os boletins deveriam ser feitos utilizando as tecnologias atuais do desenvolvimento web e seguindo as diretrizes para a melhor indexação nos motores de busca de buscadores web e seguir práticas para a melhor pontuação da plataforma referente à acessibilidade do mesmo. O trabalho a seguir foi realizado pelos alunos Jairo Pedro de Assis Junior e Fabrício Hespanhol, durante o primeiro semestre do curso.

O trabalho proposto na disciplina foi separado em diferentes páginas, cada uma contendo um conteúdo principal. Começando pela home do site, a página inicial, no canto superior direito está o logotipo da instituição, ainda na parte superior da página, porém no lado esquerdo está o menu que contém as opções de navegação pela plataforma, e por fim abaixo de tudo está um box com informações sobre o que é um boletim informativo.

Figura 2.1 - Página inicial do trabalho



Fonte: figura retirada do projeto de criação de conteúdo para web

Para o desenvolvimento da página acima foram utilizadas as seguintes linguagens: HyperText Markup Language (HTML), para criar a estrutura da página, *Cascading Style Sheets* (CSS), para fazer a estilização, e JavaScript, para a criação

da lógica entre algumas páginas e também para fazer a dinamização de componentes.

Figura 2.2- trecho de código do menu

```
<ul id="dropdown1" class="dropdown-content">
  <li><a href="boletins1.html">Boletim 1</a></li>
  <li class="divider"></li>
  <li><a href="boletins2.html">Boletim 2</a></li>
  <li class="divider"></li>
  <li><a href="boletins3.html">Boletim 3</a></li>
  <li class="divider"></li>
  <li><a href="boletins4.html">Boletim 4</a></li>
  <li class="divider"></li>
  <li><a href="boletins5.html">Boletim 5</a></li>
</ul>
```

Fonte: figura retirada do código do projeto

Na figura acima está presente um trecho de código essencial para que o menu da página principal funcione, estas pequenas linhas de código são responsáveis por criar todos os links para as páginas dos boletins. E para o funcionamento do mesmo, temos a tag *ul* que serve para criar uma lista não ordenada, uma lista que pode ter qualquer formato no site, dentro dela está a tag *li* que é onde fica alocado cada elemento da lista, dentro da tag *li* está a tag *a* que faz a ligação com a página desejada através do atributo *href* e dentro da tag *a* temos o texto que aparecerá no link esse texto serve para o usuário poder ver qual página ele estará acessando quando clicar no link. Além das tags, há também os atributos das tags, que são os textos escritos em tom mostarda, eles servem para dar alguma característica para a tag, dentro da tag *ul* estão os atributos *id* que serve para poder dar uma identificação para a tag e também o atributo *class* que disponibiliza para a tag uma característica comum para a tag.

Como já citado para a estilização das telas, usa-se a linguagem de estilização chamada CSS, que baseia-se em um esquema de cascata, onde quanto mais argumentos forem passados ao seletor de elementos mais prioritário.

Figura 2.3-trecho de código css

```
.brand-logo img{  
  background-color: white;  
  border-radius: 5px;  
  padding:5px;  
  z-index: 0;  
  margin-top:-2px;  
}
```

Fonte: código do projeto

Na primeira linha da imagem estão os seletores, que servem para selecionar uma ou mais tags, no caso, o seletor acima buscará por todas as tags `img`'s que estão dentro da classe `brand-logo`. A segunda linha serve para deixar o fundo da imagem na cor branca, por isso a palavra *white* (branco em inglês) no final da linha. Na terceira linha está a propriedade chamada `border-radius` que serve para dar uma borda ao elemento, no caso, uma borda de 5 pixels. Agora na terceira linha está a palavra *padding* que é a propriedade que serve para adicionar um espaçamento interno de também 5 pixels ao elemento. Na linha seguinte, está o comando `z-index`, que serve para dar profundidade ao elemento, no caso com o valor 0 significa que o elemento está na profundidade padrão da página, nem atrás de outros elementos nem a frente.

Na última linha de código da imagem, esta propriedade `margin-top`, que é usada para dar um espaçamento externo superior ao elemento, no caso devido ao valor negativo, o elemento está sendo puxado para cima.

Responsável pelo dinamismo de uma página, o Javascript nesse projeto foi utilizado muito para otimizar a experiência do usuário nas páginas da aplicação, sendo usado para mudar o comportamento de botões e tags, como também para criar animações acionadas com ações do usuário durante a navegação.

Figura 2.4- exemplo de código javascript

```
<script type="text/javascript">
  $(document).ready(function(){
    $('.parallax').parallax();
    $(".button-collapse").sideNav();
  });
</script>
```

Fonte: código do projeto

Primeiramente para o uso da linguagem de programação Javascript de forma inline, ou seja, dentro de um arquivo HTML, é necessário o uso da tag *script*, na linha seguinte temos a criação do evento que será iniciado assim que a página for carregada, por isso a função *ready* (preparado em inglês), sequentemente está a função que executa o efeito de parallax no elemento selecionado pelo seletor, e na última linha que realmente existe um código que executa alguma ação está o *sideNav*, que é a função que faz o efeito de *sideNav* no menu, que é o efeito de abrir uma bandeja de opções quando selecionado o menu.

Para o funcionamento da página em diferentes telas existe um tipo de consulta css que verifica qual o tamanho da tela do dispositivo que está acessando a tela, esse tipo de consulta são as Media Queries, elas funcionam como um seletor de tamanho de telas basicamente, e consegue separar um código de estilização para o tamanho da tela selecionado.

Figura 2.5- exemplo de media queries

```
@media only screen and (max-width: 600px) {
  .hide-on-small-only, .hide-on-small-and-down {
    display: none !important;
  }
}
```

Fonte: código do projeto

Primeiramente, neste código está a palavra reservada *@media* que serve para poder mostrar ao navegador que ali começa uma nova media querie, em seguida está a palavra *only* que significa que a estilização presente na media querie vai funcionar se a validação seguinte for verdadeira, em frente a palavra *only* temos *screen* que significa que vai ser levado em conta para a estilização será o tamanho

de tela de no máximo 600px, ou seja, as telas com menos de 600px terá aplicada a estilização das linhas de dentro da media querie.

Foi um trabalho muito bom, para que os alunos pudessem aprender como funciona acessibilidade em sites web e aprendessem a construir um projeto visando ajudar as pessoas com deficiência.