

08

Vantagens lado a lado

Transcrição

Continuaremos com a elaboração do e-mail marketing! Se testarmos ampliar o tamanho da tela no navegador, veremos que o espaço destinado às vantagens ficou muito grande.



Ao analisarmos o layout final no desktop, as duas tabelas devem ficar lado a lado, com um espaço no centro, que coincidiria com o tamanho do logo do Bytebank.



Está faltando limitar a largura do e-mail, o que faremos adicionando um `width` de 33% nas `td`s. Também precisaremos que o conteúdo esteja centralizado, então adicionaremos `align="center"` na `tr`.

```
<tr bgcolor="#47a237">
  <td>
```

```

<table>
  <tr width="33%">
    <td>
      <table bgcolor="blue">
        <tr>
          <td>
            
            <span class="vantagem-titulo">
              Cadastro fácil
            </span>
          <br>
          <span class="vantagem-titulo">
            Ingresso simples, sem burocracia, sem análises extensas e com muita rapidez.
          </span>
        </tr>
      </table>
    </td>
  </tr>
</table>
<!-- ... -->

```



Com dois terços do espaço ocupados, ficaram faltando 33% da largura para ficar bem distribuído. Para este ajuste, logo abaixo da `td` de "Cadastro fácil", criaremos outra `td` com `width` igual a 34% - para que o total seja de 100%.

```

tr bgcolor="#47a237">
  <td>
    <table>
      <tr width="33%">
        <td>
          <table bgcolor="blue">
            <tr>
              <td>
                
                <span class="vantagem-titulo">
                  Cadastro fácil
                </span>
              <br>
              <span class="vantagem-titulo">
                Ingresso simples, sem burocracia, sem análises extensas e com muita rapidez.
              </span>
            </tr>
          </table>
        </td>
      </tr>
    </table>
  </td>
  <td>
    <table>
      <tr width="34%">
        <td>
          <table bgcolor="blue">
            <tr>
              <td>
                
                <span class="vantagem-titulo">
                  Cadastro fácil
                </span>
              <br>
              <span class="vantagem-titulo">
                Ingresso simples, sem burocracia, sem análises extensas e com muita rapidez.
              </span>
            </tr>
          </table>
        </td>
      </tr>
    </table>
  </td>
</tr>

```

```

</td>
<td width="34%">
  &nbsp;
</td>

```

Adicionaremos o ` ` para indicar a existência do espaço entre as tabelas.



Apesar da inclusão de espaço, ainda existe a sensação de vazio no layout. Se queremos criar um design responsivo, é preciso trabalhar com a porcentagem. Porém, em alguns momentos, utilizaremos pixels, ou seja, unidades fixas. Para obtermos o resultado desejado, adicionaremos o valor `200` para `width`, resultado da divisão de `600` por `3`.

```

<tr>
  <td bgcolor="#47a138" align="center">
    
  </td>
</tr>

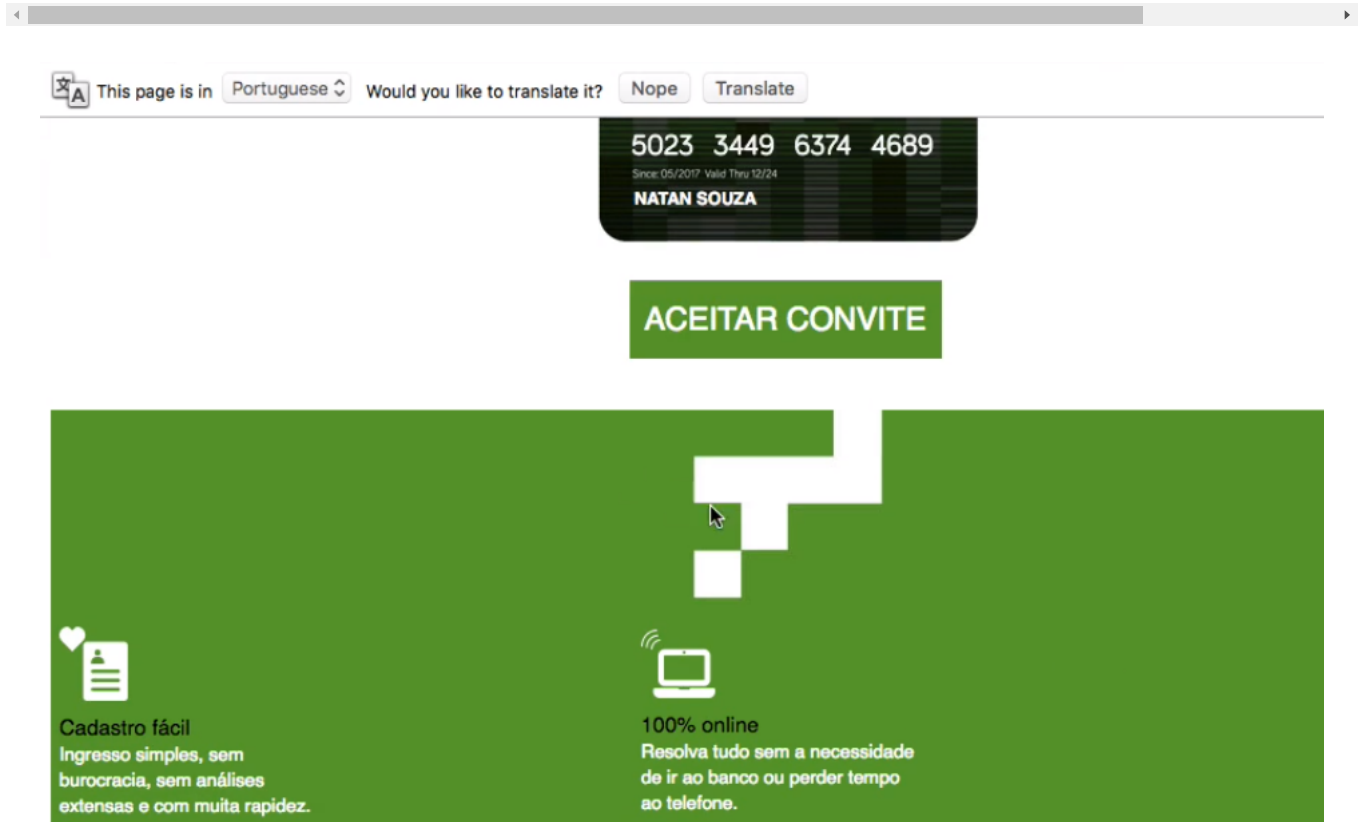
<tr bgcolor="#47a138">
  <td align="center">
    <table>
      <tr>
        <td width="200">
          <table>
            <tr>
              <td>
                
                <span class="vantagem-titulo">
                  Cadastro fácil
                </span>
                <br>
                <span class="vantagem-texto">
                  Ingresso simples, sem burocracia, sem análises extensas e com m
                </span>
                <br>
              </td>
            </tr>
          </table>
        </td>
      </tr>
    </table>
  </td>
  <td width="133px">

```

```

    &nbsp;
  </td>
<!-- ... -->

```



O tamanho da `td` foi reduzido, porém aparentemente o `align="center"` não está funcionando. Isso porque o mais apropriado seria adicioná-lo nas células. Faremos a alteração no `td` acima da primeira célula, e além disso diminuiremos o valor do `width` para 150 :

```

<tr bgcolor="#47a138">
  <td align="center">
    <table>
      <tr>
        <td width="200">

```

O espaço entre as tabelas foi reduzido no layout:



Reduziremos o valor do width na td do meio para 135 :

```
<td width="135px">
    &nbsp;
</td>
```

Quando formos trabalhar com a versão mobile, as vantagens deverão ser exibidas uma sobre a outra.

Um ponto que devemos levar em consideração é o fato de estarmos trabalhando com a largura fixa, algo interessante quando trabalhamos com design responsivo. O ideal em um caso como o nosso é trabalhar com a porcentagem de 33% - mas como deixaremos a largura com um valor fixo? Poderemos definir isto no CSS. Para simplificarmos o uso do seletor, definiremos que trabalharemos com td , que é filha td :

```
<style>

td td {
    background: red;
}

img {
    display: block;
}

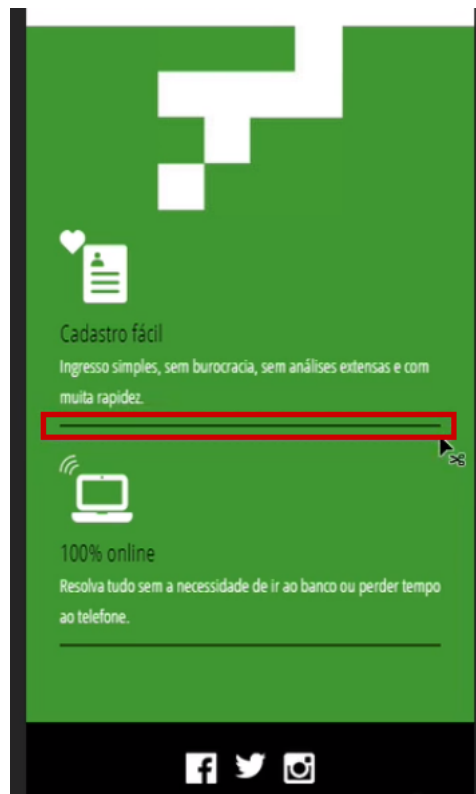
<!-- ... -->
```



Vamos configurá-la com uma largura máxima de 133px , pois queremos que e-mail expanda na largura, porém até certo limite.



O espaço entre as tabelas coincide com o espaço do logo, além de corresponder com a versão final do layout. Mais adiante, faremos com que as vantagens quebrem em duas linhas na versão mobile. Teremos que adicionar as linhas exibidas na versão final.



Usaremos uma técnica antiga para criá-las, adicionando o `<hr>` (*horizontal row*) abaixo do ``, com o texto de descrição da vantagem, que já estará responsiva. Mas para deixá-la com a cor apropriada do layout final, incluiremos o CSS e as propriedades relativas à sua borda.

```
<tr bgcolor="#46a237">
  <td align="center">
    <table>
      <tr>
        <td width="33%">
          <table>
            <tr>
              <td>
                
                <span class="vantagem-titulo">
                  Cadastro fácil
                </span>
                <br>
                <span class="vantagem-titulo">
                  Ingresso simples, sem burocracia, sem análises extensas e com muita rapidez
                </span>
                <br>
                <hr style="border-color: #195a0e;">
              </td>
            </tr>
          </table>
        </td>
        <td width="33%">
          &nbsp;
        </td>
        <td width="33%">
          <table>
            <tr>
              <td>
                
```

```

        <span class="vantagem-titulo">
            100% online
        </span>
        <br>
        <span class="vantagem-texto">
            Resolva tudo sem a necessidade de ir ao banco ou perder tempo ao telefone.
        </span>
        <br>
        <hr style="border-color: #195a0e">
    </td>
</tr>
</table>
</td>
<!-- ... -->

```

Incluiremos o `style` nas duas tags `<hr>` s:



Nós escolhemos um tom escuro de verde para a linha, mas você pode testar outras cores com a ferramenta *Inspect*. Neste caso, a cor já havia sido definida, então devemos respeitar a decisão do designer.

Vale lembrar que assim como trabalhamos com uma borda para a tag `<hr>`, poderíamos criar uma borda ao redor de toda a vantagem. Realizaremos um teste apenas para mostrar como seria isso, adicionando a classe `vantagem` no CSS.

```

.cta {
    background-color: #47a138;
    color: #ffffff;
    display: inline-block;
    font-family: sans-serif;
    font-size: 22px;
    line-height: 55px;
    text-align: center;
    text-decoration: none;
    width: 219px;
    -webkit-text-size-adjust: none;
}

.vantagem {
    border-bottom: 2px solid #195a0e
}

```


Em seguida, em cada uma das tags `td`s que está representando as vantagens, incluiremos a classe `vantagem`. No trecho relacionado ao "100% online":

```
<td width="33%" class="vantagem">
  <table>
    <tr>
      <td>
        
        <span class="vantagem-titulo">
          100% online
        </span>
        <br>
        <span class="vantagem-texto">
          Resolva tudo sem a necessidade de ir ao banco ou perder tempo ao telefone.
        </span>
        <br>
        <hr style="border-color: #195a0e">
      </td>
    </tr>
  </table>
</td>
```



A linha criada é bastante parecida e, para fins práticos, serviria também. Mas, novamente, corremos o risco do email client não dar suporte ao `border-bottom`, portanto deixaremos essa linha do código comentada.

```
.vantagem {
  /*border-bottom: 2px solid #195a0e*/
}
```

Em seguida, daremos continuidade nesta parte do e-mail marketing responsivo.

Nesta aula, vimos a parte de *call to action*, incluindo o link em uma parte específica. Vimos como criar tabelas dentro de tabelas (conceito de *multigridd*), o que pode deixar o nosso código um pouco "assustador".

Quando o colapsamos e recolhemos alguns de seus trechos, vimos que a abordagem faz sentido. Por último, usamos a tag `hr`, bastante antiga, e para mudarmos suas propriedades visuais aplicamos estilos de CSS relacionados à bordas.

