

Web Mobile

Desenvolvido por:

André Lima 10410280

Alan Ribeiro do Carmo 10428496

Giovanni Suppo 10438719

Henrique Duraó 1043806

São Paulo, abril de 2025

Ideação

Nesse projeto, vamos fazer um portfólio sobre um jogador de futebol, nesse caso, pensamos no jogador Cristiano Ronaldo. Esse projeto, inicialmente, será um desenvolvimento de uma página com HTML, CSS e Javascript. Será uma página com o intuito de apresentar a vida, carreira e características desse jogador.

Tutorial

Inicialmente, para criar um projeto em HTML, é necessário informar no código que é um HTML, para funcionar de forma correta, então sempre ao início de um código HTML, é necessário iniciar com o `<!DOCTYPE html>`.

Antes de inserir os códigos, também é necessário que estejam dentro das tags html. É necessário abrir e fechar o `<html>`.

```
1  <!DOCTYPE html>
2  <html>
```

```
42 </html>
```

Após isso, podemos começar a colocar as informações de nosso html, para isso, iniciamos uma tag `<head>`, que é mais para informar os tipos de dados, título da página e algumas informações importantes. Nesse caso, foi inserido o meta charset UTF-8, que é um padrão para codificar caracteres Unicode. Também há o meta da linha 5 que é necessário num código responsivo.

Abaixo, na linha 6, foi colocado o título que ficará na guia do navegador.

```
3  <head>
4    <meta charset = "UTF-8">
5    <meta name = "viewport" content = "width=device-width, initial-scale = 1.0">
6    <title>Projeto 1</title>
7  </head>
```

Após isso, podemos iniciar os códigos do que será apresentado na página, mas para isso, é necessário colocar todo o código dentro das tags body, que é o corpo da página.

```
9  <body>
```

```
41 </body>
```

Logo após, iniciamos o código. Adicionamos a tag `<header>` para indicar o cabeçalho da página, as informações que ficarão na parte superior.

Dentro do header, colocamos um título, nesse caso, colocamos a tag `<h1>`, mas poderia ir de h1 até h6.

Após isso, há a tag nav, que indica um campo de navegação, nesse campo que iremos colocar os links para as próximas páginas.

O `` indica uma lista de itens sem uma ordem rígida. Os `` são para colocar cada item da lista. A tag `<a>` com um `href`, indica onde será direcionado após clicar nesse link, nesse caso está o `"#"` para continuar na mesma página.

Após isso, fechamos todas as tags.

```
9      <body>
10      <header>
11      <h1>Projeto 1</h1>
12      <nav>
13      <ul>
14      <li><a href = "#">Home</a></li>
15      <li><a href = "#">Página 2</a></li>
16      <li><a href = "#">Página 3</a></li>
17      <li><a href = "#">Sobre</a></li>
18      </ul>
19      </nav>
20      </header>
```

Após isso, foi criado uma section, uma seção, onde foi colocado algumas outras tags do html, nesse caso, utilizamos articles para, no futuro, colocarmos textos futuramente. Nesse caso há parágrafos indicando os artigos.

```
22      <section>
23      <article>
24      <p>Artigo 1</p>
25      </article>
26
27      <article>
28      <p>Artigo 2</p>
29      </article>
30
31      <article>
32      <p>Artigo 3</p>
33      </article>
34
35      </section>
36
37
```

No final, colocamos um footer, que é o rodapé da página. Ficará na parte inferior da página com algumas infos.

```
38      <footer>
39      <p>2025 - Todos os direitos reservados</p>
40      </footer>
```

Nessa 2º parte do projeto, adicionamos conteúdos no HTML e estilizamos a página com CSS.

HTML

Logo de início, no head, colocamos a tag <link> informando que é uma estilização com CSS e colocamos o nome do arquivo que está com a estilização.

```
<head>
  <meta charset = "UTF-8">
  <meta name = "viewport" content = "width=device-width, initial-scale = 1.0">
  <title>Cristiano Ronaldo</title>
  <link rel = "stylesheet" type = "text/css" href = "style.css"/>
</head>
```

Dentro do body, começamos fazendo algumas alterações no header, adicionamos uma imagem, que é o logo do site, então o "src" indica o arquivo da imagem e o "alt" é uma descrição da imagem. Separamos em duas navs para ficar uma à direita e outra à esquerda da imagem. Colocamos classes nas navs e nas ul's para ficar mais fácil de estilizá-las no CSS. Os links redirecionam para partes específicas da mesma página.

```
<header>
  <nav class = "nav1">
    <ul class = "ul1">
      <li><a href = "#secao">Quem é</a></li>
      <li><a href = "#artigo2">Carreira</a></li>
    </ul>
  </nav>
  <img src = "logo.png" alt = "logo">
  <nav class = "nav2">
    <ul class = "ul1">
      <li><a href = "#artigo3">Títulos</a></li>
      <li><a href = "#artigo4">Curiosidades</a></li>
    </ul>
  </nav>
</header>
```

Em seguida, começamos a section com os artigos. Em cada article colocamos os ids para especificar no CSS. Em todos os articles há primeiro uma imagem, depois uma section para separar a parte do texto. Nesse caso, há primeiro um h1 com o título daquele article, depois um parágrafo com o texto.

```

<section id = "secao">

  <article id = "artigo1">
    <img src = "cr7_1.jpg" alt = "cr7_inicio">
    <section class = "texto">
      <h1>Quem é Cristiano Ronaldo?</h1>
      <p>Cristiano Ronaldo dos Santos Aveiro, nascido em 5 de fevereiro de 1985, na Ilha da Madeira, Portugal, é um dos maiores jogadores de futebol da história. Ele se destacou desde cedo pelo talento e dedicação, tornando-se referência mundial em técnica, velocidade e finalização. Com uma carreira vitoriosa em clubes europeus e na Seleção Portuguesa, Ronaldo coleciona recordes e prêmios individuais. Além do futebol, é uma personalidade influente no mundo dos negócios e um ícone global do esporte.</p>
    </section>
  </article>

```

No 2º article, mantemos uma estrutura parecida, mas para separar os parágrafos, achamos melhor colocar várias tags <p>. Além disso, como o último parágrafo ficava muito colado no final do article, foi colocada uma id, para ajustar isso no CSS.

```

<article id = "artigo2">
  <img src = "cr7_2.jpg" alt = "CR7_united">
  <section class = "texto">
    <h1>Carreira</h1>
    <p>Cristiano Ronaldo começou sua trajetória no futebol no Andorinha, um pequeno clube da Ilha da Madeira. Aos 12 anos, foi para a base do Sporting CP, em Lisboa, onde desenvolveu sua técnica e rapidamente chamou a atenção dos grandes clubes europeus.</p>
    <p>Em 2003, após se destacar em um amistoso contra o Manchester United, foi contratado pelo clube inglês aos 18 anos. No United, sob comando de Alex Ferguson, Ronaldo evoluiu fisicamente e se tornou um dos melhores jogadores do mundo, conquistando três Premier League, uma Champions League e sua primeira Bola de Ouro em 2008.</p>
    <p>Em 2009, foi transferido para o Real Madrid por €94 milhões, um recorde na época. No clube merengue, CR7 viveu o auge da carreira, tornando-se o maior artilheiro da história do time com 450 gols. Lá, conquistou quatro Champions League, duas La Ligas e inúmeros títulos individuais, incluindo quatro Bolas de Ouro.</p>
    <p>Após nove anos no Real Madrid, transferiu-se para a Juventus, em 2018, por €100 milhões. Na Itália, manteve sua regularidade e ajudou a equipe a vencer dois Campeonatos Italianos e a Copa da Itália.</p>
    <p>Em 2021, retornou ao Manchester United, mas a passagem foi conturbada devido a problemas com a diretoria e comissão técnica. Em 2022, deixou o clube e assinou com o Al-Nassr, da Arábia Saudita, tornando-se um dos maiores nomes do futebol asiático.</p>
    <p id = "ultimop">Pela Seleção Portuguesa, Ronaldo estreou em 2003 e desde então se tornou o maior artilheiro da história do time nacional. Conquistou a Eurocopa de 2016 e a Liga das Nações da UEFA em 2019, consolidando seu nome como o maior jogador da história de Portugal.</p>
  </section>
</article>

```

No 3º article, foi feito de uma forma um pouco diferente, pois colocamos o <h1> com o título e um <p> com o parágrafo, mas criamos uma lista com os títulos do Cristiano Ronaldo, então, criamos um e nele colocamos os times que ele passou nos principais li's, e para criar um sub-item na lista (que são os títulos), é feito dessa forma: dentro do cria-se um e dentro desse é criado outro pra criar uma sub-lista e dentro dessa sub-lista é criado outro , onde estão os títulos.

```

<article id = "artigo3">
  <img src = "cr7_3.webp" alt = "CR7_united">
  <section class = "texto">
    <h1>Títulos:</h1>
    <p>Cristiano Ronaldo tem uma carreira marcada por títulos nacionais, internacionais e prêmios individuais. Ele venceu títulos importantes em todos os clubes pelos quais passou:</p>
    <ul class = "ul3">
      <li>Sporting CP 🏆</li>
      <li>
        <ul>
          <li>Supertaça de Portugal (2002)</li>
        </ul>
      </li>
      <li>Manchester United 🏆🏆🏆</li>
      <li>
        <ul>
          <li>Premier League (2006-07, 2007-08, 2008-09)</li>
          <li>FA Cup (2003-04)</li>
          <li>Copa da Liga Inglesa (2005-06, 2008-09)</li>
          <li>Supercopa da Inglaterra (2007, 2008)</li>
          <li>Champions League (2007-08)</li>
          <li>Mundial de Clubes da FIFA (2008)</li>
        </ul>
      </li>
    </ul>
  </li>

```

Foi feito dessa maneira para todos os times que o Cristiano passou e também para a seleção portuguesa. No final, há um parágrafo com mais informações.

```

      <li>Real Madrid 🏆🏆🏆🏆🏆🏆</li>
      <li>
        <ul>
          <li>La Liga (2011-12, 2016-17)</li>
          <li>Copa do Rei (2010-11, 2013-14)</li>
          <li>Supercopa da Espanha (2012, 2017)</li>
          <li>Champions League (2013-14, 2015-16, 2016-17, 2017-18)</li>
          <li>Supercopa da UEFA (2014, 2017)</li>
          <li>Mundial de Clubes da FIFA (2014, 2016, 2017)</li>
        </ul>
      </li>
      <li>Juventus 🏆🏆🏆</li>
      <li>
        <ul>
          <li>Serie A (2018-19, 2019-20)</li>
          <li>Supercopa da Itália (2018, 2020)</li>
          <li>Copa da Itália (2020-21)</li>
        </ul>
      </li>
      <li>Seleção Portuguesa pr</li>
      <li>
        <ul>
          <li>Eurocopa (2016)</li>
          <li>Liga das Nações da UEFA (2019)</li>
        </ul>
      </li>
    </ul>
    <p>Além dos títulos coletivos, Cristiano Ronaldo possui cinco Bolas de Ouro (2008, 2013, 2014, 2016, 2017), sendo um dos jogadores mais premiados da história.</p>
  </section>
</article>

```

Em seguida, há o article 4, que fizemos seguindo o exemplo do article 3, em forma de lista. Nesse article colocamos as curiosidades sobre o CR7. Em seguida acaba a </section> que contém os articles.

```

<article id = "artigo4">
  <img src = "cr7_5.webp" alt = "CR7_alnassr">
  <section class = "texto">
    <h1>Curiosidades:</h1>
    <ul class = "ul2">
      <li>Seu nome "Ronaldo" foi escolhido por causa do ex-presidente dos EUA, Ronald Reagan, que era o ídolo do pai dele.
    </li>
      <li>Ele é o primeiro jogador da história a marcar em cinco edições diferentes da Copa do Mundo (2006, 2010, 2014, 2018 e 2022).
    </li>
      <li>É o maior artilheiro da Champions League, com mais de 140 gols na competição.
    </li>
      <li>Mantém uma dieta extremamente rigorosa e tem um percentual de gordura menor do que muitos atletas olímpicos.
    </li>
      <li>Já doou milhões de euros para hospitais e instituições de caridade e recusou contratos publicitários de empresas que promovem bebidas alcoólicas ou jogos de azar.
    </li>
      <li>É o atleta mais seguido do mundo nas redes sociais, com mais de 600 milhões de seguidores no Instagram.
    </li>
    </ul>
  </section>
</article>
</section>

```

Por último, está o footer, o rodapé da página. Nele está as informações dos componentes do grupo. Adicionamos os parágrafos com os nomes e RA's.

```

<footer>
  <p> Trabalho desenvolvido por:</p>
  <p> Alan Ribeiro do Carmo - 10428496 </p>
  <p> André Lima - 10410280 </p>
  <p> Giovanni Suppo - 10438719 </p>
  <p> Henrique Durão - 10438068 </p>
</footer>
</body>
</html>

```

Com o fechamento das tags body e html, se encerra o arquivo HTML do trabalho.

CSS

Iniciamos a estilização da nossa página com os elementos que estão dentro da NAV.

Primeiramente, estilizamos o primeiro , que é o que vai ter os links para as partes da página. Primeiro colocamos o "align-items: center" para centralizar os na página. O "display: flex" para os itens ficarem flexíveis e poderem ser movimentados de acordo com a configuração do CSS. Finalizando com o "flex-wrap: nowrap", para que os itens continuem alinhados e não quebrem para baixo.

Depois, estilizamos os dentro desse . Colocamos um padding para ter um preenchimento interno e não ficarem muito colados um no outro e para o texto não ficar colado na borda do componente. O "list-style-type: none" para que não fique estilizado aquele

texto, para não ter aparência de lista. Depois colocamos uma transition, para que quando for passado o mouse acima, aparecer uma cor de fundo, com uma sombra.

No <a> foi tirada a decoração e foi adicionada a cor preta, para que não tenha aparência de link.

Depois, colocamos o “.ul1 li: hover”, para que os dessa de classe “ul1” tenham certos efeitos quando for passado o mouse acima: cor de fundo e sombra (que estão na transition citada anteriormente) e uma borda de 15 pixels.

```
1  .ul1{
2      align-items: center;
3      display : flex;
4      flex-wrap: nowrap;
5  }
6
7  .ul1 li{
8      padding: 5px;
9      list-style-type: none;
10     transition: background-color 0.3s ease, box-shadow 0.3s ease;
11 }
12
13 a{
14     text-decoration: none;
15     color: black;
16 }
17
18
19 .ul1 li:hover{
20     background-color: rgb(226, 220, 220);
21     border-radius: 5px;
22     box-shadow: 10px 10px 20px rgba(59, 52, 52);
23 }
24 }
```

Após isso, pensamos em alterar mais coisas:

Para a tag html, colocamos o “scroll-behavior: smooth”, para que quando o usuário clicar nos links do , a página abaixe lentamente até o lugar desejado da página.

Depois foram feitos alguns ajustes na imagem do logo. Ela terá 5% de largura e altura e foi tirado o espaçamento de margin e padding.

Logo após, foram feitos os ajustes na section principal, onde estão os articles. Os itens foram alinhados ao centro, o display ficou flexível para os itens poderem ser movimentados. O “flex-wrap: wrap” para que os itens quebrem para baixo caso não couberem no espaço e estão direcionados em forma de coluna. Depois foi colocado margin para que a section se afaste um pouco do header e footer e o scroll-margin-top: 200px para que quando a tela rolar, tenha essa margin de 200px para o topo.

Em seguida, fizemos uma estilização geral dos articles com largura de 80%, cor de fundo dourada, margin de 10px, uma borda e uma sombra.

```
39  html{
40      scroll-behavior: smooth;
41  }
42
43  img{
44      width: 5%;
45      height: 5%;
46      padding: 0;
47      margin: 0;
48  }
49
50  section{
51      align-items: center;
52      display: flex;
53      flex-wrap: wrap;
54      flex-direction: column;
55      margin: 10px;
56      margin-top: 100px;
57      scroll-margin-top: 200px;
58  }
59
60  article{
61      width: 80%;
62      background-color: #d4af37;
63      margin: 10px;
64      border-radius: 15px;
65      box-shadow: 10px 10px 20px #444;
66  }
67
```

Então, começamos a estilizar os articles um por um.

O "flex-direction: row" para que a imagem fique à esquerda e os textos à direita. O display está flexível, o justify-content: space-around para que os itens sejam distribuídos com o espaçamento igual. Os itens estão alinhados ao centro e esse article tem uma margem na parte de cima.

Depois estilizamos a imagem, com largura e altura de 200px e uma margin e um padding.

O texto desse article foi estilizado de forma que o parágrafo e o título fiquem em coluna e o parágrafo não tem espaçamento interno, mas tem um espaçamento externo na parte de baixo de 2px.

```

68 #artigo1{
69     flex-direction: row;
70     display: flex;
71     justify-content: space-around;
72     align-items: center;
73     margin-top: 50px;
74 }
75
76 #artigo1 img{
77     margin: 5px;
78     padding: 20px;
79     width: 200px;
80     height: 200px;
81 }
82
83 .texto{
84     display: flex;
85     flex-direction: column;
86     padding: 0;
87     margin: 0;
88 }
89
90 .texto p{
91     padding: 0;
92     margin-bottom: 2px;
93 }
94

```

No 2º article, há uma estilização parecida com a do 1º, mas com algumas alterações, o parágrafo tem um espaçamento interno à direita de 20px e o último parágrafo tem uma margin e um padding para evitar que o texto fique muito colado na borda do article.

```

96 #artigo2{
97     flex-direction: row;
98     display: flex;
99     justify-content: space-around;
100     align-items: center;
101 }
102
103 #artigo2 img{
104     margin: 5px;
105     padding: 20px;
106     width: 200px;
107     height: 200px;
108 }
109
110 #artigo2 p{
111     padding-right: 20px;
112 }
113
114 #ultimop{
115     margin-bottom: 10px;
116     padding-bottom: 5px;
117 }
118

```

O article 3 e 4 foram estilizados da mesma forma que os anteriores.

```
119 #artigo3{
120     flex-direction: row;
121     display: flex;
122     justify-content: space-around;
123     align-items: center;
124 }
125
126 #artigo3 img{
127     margin: 5px;
128     padding: 20px;
129     width: 200px;
130     height: 200px;
131 }
132
133 #artigo3 p{
134     padding-right: 20px;
135 }
136
137 #artigo4{
138     flex-direction: row;
139     display: flex;
140     justify-content: space-around;
141     align-items: center;
142 }
143
144 #artigo4 img{
145     margin: 5px;
146     padding: 20px;
147     width: 200px;
148     height: 200px;
149 }
```

Assim como o article 5.

```
151 #artigo5{
152     flex-direction: row;
153     display: flex;
154     justify-content: space-around;
155     align-items: center;
156 }
157
158 #artigo5 img{
159     margin: 5px;
160     padding: 20px;
161     width: 200px;
162     height: 200px;
163 }
164
165 #artigo5 p{
166     padding-right: 20px;
167 }
168
```

Foram feitas algumas estilizações no body, mantendo o padrão original, para que não haja nenhum bug ou que fosse impactado por outros componentes.

```
170 body{
171     margin: 0;
172     padding: 0;
173     background-color: #fff;
174 }
175
```

Editamos também o rodapé. Colocamos uma cor de fundo cinza, uma sombra, o display está flexível e os itens podem quebrar para a linha abaixo. Uma forma para garantir que o footer ficará fixo no final da página é colocar o “position: absolute”, para garantir que ficará absolutamente naquela posição, também colocamos o width: 100% para garantir que ocupará toda a largura da página e um espaçamento interno zerado.

Também colocamos um espaçamento interno à esquerda nos parágrafos do footer para que não fique totalmente colados no canto da tela.

```
176 footer{
177     background-color: #666;
178     position: absolute;
179     width: 100%;
180     padding: 0;
181     display: flex;
182     flex-wrap: wrap;
183     box-shadow: 5px 5px 20px #333;
184 }
185
186 footer p{
187     padding-left: 10px;
188 }
```

Após isso foram feitas algumas alterações gerais na página que achamos necessário para que ficasse com um design mais agradável e funcional.

Foi colocada a fonte Avantgarde em todos os tipos de texto da página e os ul2 e ul3 ficaram flexíveis e alocados em coluna.

```

190 h1{
191     margin: 0;
192     padding: 0;
193 }
194
195 h1{
196     font-family: Avantgarde, sans-serif;
197     padding: 10px;
198 }
199
200 p{
201     font-family: Avantgarde, sans-serif;
202 }
203
204 .ul2{
205     display: flex;
206     flex-direction: column;
207 }
208
209 .ul2 li{
210     font-family: Avantgarde, sans-serif;
211 }
212
213 .ul3{
214     display: flex;
215     flex-direction: column;
216 }
217
218 .ul3 li{
219     font-family: Avantgarde, sans-serif;
220 }
221

```

Nesse momento, começa a estilização para outros tamanhos de tela, tornando o site responsível. Para isso é necessário colocar o @media e informar a largura de tela em que será feitas essas alterações, que serão alocadas dentro desse media, nesse caso, para dispositivos com até 768px de largura.

Alteramos inicialmente a logo do site e todos os articles passaram a ter um espaçamento interno de 3%

```

222 @media (max-width: 768px) {
223     img{
224         width: 50px;
225         height: 50px;
226     }
227
228     article{
229         padding: 3%;
230     }
231

```

Após isso, todos os articles receberam essa configuração:

```

233     #artigo1{
234         display: flex;
235         flex-direction: column;
236         align-items: center;
237         width: 90%;
238     }
239
240     #artigo2{
241         display: flex;
242         flex-direction: column;
243         align-items: center;
244         width: 90%;
245     }
246
247     #artigo3{
248         display: flex;
249         flex-direction: column;
250         align-items: center;
251         width: 90%;
252     }
253
254     #artigo4{
255         display: flex;
256         flex-direction: column;
257         align-items: center;
258         width: 90%;
259     }
260
261     #artigo5{
262         display: flex;
263         flex-direction: column;
264         align-items: center;
265         width: 90%;
266     }

```

Todos os articles tem os componentes flexíveis, em coluna, alinhados ao centro e tem largura de 90%. Essa estilização foi feita para cada article separado por conta de ter possibilidade de ser alterado depois individualmente. Dessa forma, a imagem fica acima do texto quando for visualizado por tablet ou celular.

Após isso, o ul1 e seus componentes foram estilizados da mesma forma que no desktop, mas foram acrescentados no @media para caso seja necessária alguma modificação. Durante os testes foi necessário.

```

270     .ul1{
271         align-items: center;
272         display : flex;
273         justify-content: center;
274         padding: 0;
275         margin:0;
276     }
277
278     .ul1 li{
279         list-style-type: none;
280         transition: background-color 0.3s ease, box-shadow 0.3s ease;
281     }
282
283     a{
284         text-decoration: none;
285         color: □black;
286     }
287

```

Após isso, as imagens de cada article tiveram essa configuração, onde a largura ocupa 80%, a altura ocupa 20% e há uma margin de 5px e um padding de 20px para que não fique muito colado ao texto ou à borda do article. Também há uma configuração para cada imagem por conta que a imagem pode ser alterada futuramente e será mais fácil a configuração dessa imagem.

```

288     #artigo1 img{
289         margin: 5px;
290         padding: 20px;
291         width: 80%;
292         height: 20%;
293     }
294
295     #artigo2 img{
296         margin: 5px;
297         padding: 20px;
298         width: 80%;
299         height: 20%;
300     }
301
302     #artigo3 img{
303         margin: 5px;
304         padding: 20px;
305         width: 80%;
306         height: 20%;
307     }
308
309     #artigo4 img{
310         margin: 5px;
311         padding: 20px;
312         width: 80%;
313         height: 20%;
314     }
315
316     #artigo5 img{
317         margin: 5px;
318         padding: 20px;
319         width: 80%;
320         height: 20%;
321     }
322 }

```

Observando que a página pode ficar com a visualização um pouco comprometida quando a largura é abaixo de 390px, fizemos mais uma configuração, onde a imagem tem um tamanho diferente e uma margin na parte de cima. Além disso, percebemos que a fonte dos <a> estava muito pequena, então foi colada essa configuração de “font-size: 100%”. Depois disso, o <ul1> foi modificado de forma que os quebrem e fiquem abaixo um do outro, em coluna.

```
324 @media (max-width: 390px) {
325     img{
326         width: 20%;
327         height: 20%;
328         margin-top: 5%;
329     }
330     .ul1 li a{
331         font-size: 100%;
332     }
333
334     .ul1{
335         align-items: center;
336         display : flex;
337         justify-content:center;
338         flex-direction: column;
339
340         padding: 0;
341         margin:0;
342     }
343 }
```

-----3ª Parte-----

Na 3ª parte do projeto, foi adicionada uma função com javascript.

Inicialmente criamos um arquivo chamado script.js, onde adicionamos o código e referenciamos ele no <body> do html:

```
157 <script src = script.js></script>
```

A funcionalidade que adicionamos é um botão “mostrar mais” em dois articles, para expandir o texto. Para isso funcionar, adicionamos o botão e classes no HTML com o nome “hidden”, para que o javascript entenda que essa parte do texto que terá essa funcionalidade.

```
143 <button class="mostrar-mais">Mostrar menos</button>
```

```
70 <ul class="ul3 hidden">
```



```

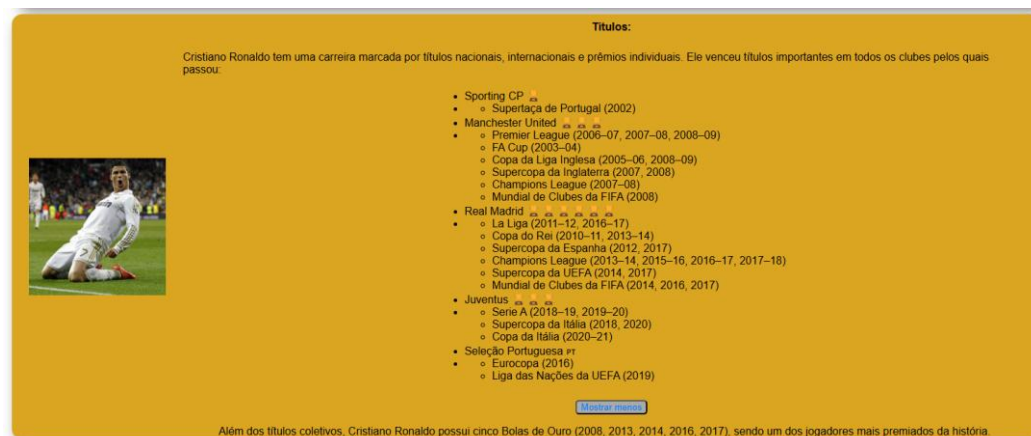
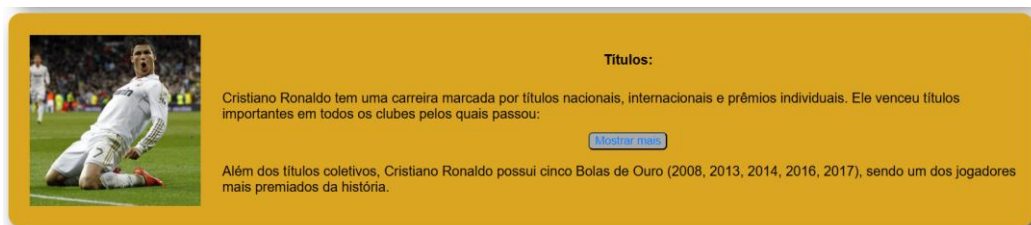
1 document.querySelectorAll('.mostrar-mais').forEach(button => {
2     button.addEventListener('click', function() {
3         const list = button.previousElementSibling;
4
5         list.classList.toggle('hidden');
6
7         if (list.classList.contains('hidden')) {
8             button.textContent = 'Mostrar mais';
9         } else {
10            button.textContent = 'Mostrar menos';
11        }
12    });
13 });

```

Nesse código, é selecionado todos os elementos com a classe “mostrar-mais” e para cada botão nessa classe é executado o código da função.

Depois, há um evento de click, em que executa um código que pega o elemento anterior ao botão (previousElementSibling) e adiciona ou remove a classe “hidden” da lista, ficando visível ou invisível, depois, verifica se a lista com a classe “hidden” está escondida ou não para alterar o nome do botão.

Visualmente:



Na 4ª parte do projeto, implementamos o botão para alternar entre modo escuro e modo claro, para isso, fizemos as seguintes adições no projeto:

Adicionamos uma classe onde ficará esse botão, dentro dele, colocamos um input do tipo checkbox para selecionar ou não selecionar o modo escuro, podem essa checkbox ficará oculta. Depois, colocamos uma label, que é o espaço em que ficará o botão e os símbolos, onde foi adicionado duas tags <i> que são apenas para referenciar a imagem que está sendo adicionada no documento a partir do script abaixo, onde foi retirado de um site que contém algumas imagens com símbolos, lá foram retirados os símbolos do sol e lua do botão.

```
25     </nav>
26     <section class = "botao">
27         <input type = "checkbox" class = "checkbox" id = "chk"/>
28         <label class = "label" for = "chk">
29             <i class = "fas fa-moon"></i>
30             <i class = "fas fa-sun"></i>
31             <div class = "ball"></div>
32         </label>
33     </section>
34 </header>

166
167     <script src="script.js"></script>
168     <script src="https://kit.fontawesome.com/88a5fa2efb.js" crossorigin="anonymous"></script>
169
170 </body>
```

No CSS, estilizamos alguns elementos com um “.dark” para referenciar no javascript.

```
155
156 body.dark{
157     background: #292c35;
158 }
159
160 body.dark article{
161     background-color: rgb(59, 52, 52);
162     box-shadow: 3px 3px 20px rgb(179, 172, 172);
163     color: white;
164 }
165
166 body.dark header{
167     background-color: #685405;
168 }
169
170 body.dark a{
171     color: white;
172 }
173
```

Depois, é necessário estilizar o botão, foi feito dessa maneira:

```
239
240 .botao{
241     margin: 0;
242     padding: 0;
243     float: right;
244     display: flex;
245     align-items: center;
246     justify-content: center;
247     width: 50px;
248     height: 50px;
249 }
250
251 .checkbox{
252     opacity: 0;
253     position: absolute;
254 }
```

O elemento checkbox fica invisível, mas ainda acessível pela label e fica fora do fluxo do layout

Depois foi estilizado a label:

```
255
256 .label{
257     background-color: #111;
258     border-radius: 50px;
259     cursor: pointer;
260     display: flex;
261     align-items: center;
262     justify-content: space-between;
263
264     padding: 5px;
265     position: relative;
266     height: 26px;
267     width: 50px;
268
269     transform: scale(0.8);
270 }
271
272 .label .ball{
273     background-color: #fff;
274     border-radius: 50%;
275     position: absolute;
276     top: 2px;
277     left: 2px;
278     height: 30px;
279     width: 30px;
280
281     translate: translateX(0px);
282     transition: transform 0.2s linear;
283 }
```

Os elementos dentro da label ficam separados lado a lado com space-between, e foi ajustada a questão da cor e tamanhos.

Depois, foi estilizada a bolinha dentro da label, o translate representa que esse elemento irá se mexer e ele tem uma pequena transição para se mover aos poucos.

```
285 .checkbox:checked + .label .ball{
286   transform: translate(24px);
287 }
288
289 .fa-moon{
290   color: #f1c40f;
291 }
292
293 .fa-sun{
294   color: #f39c12;
295 }
296
```

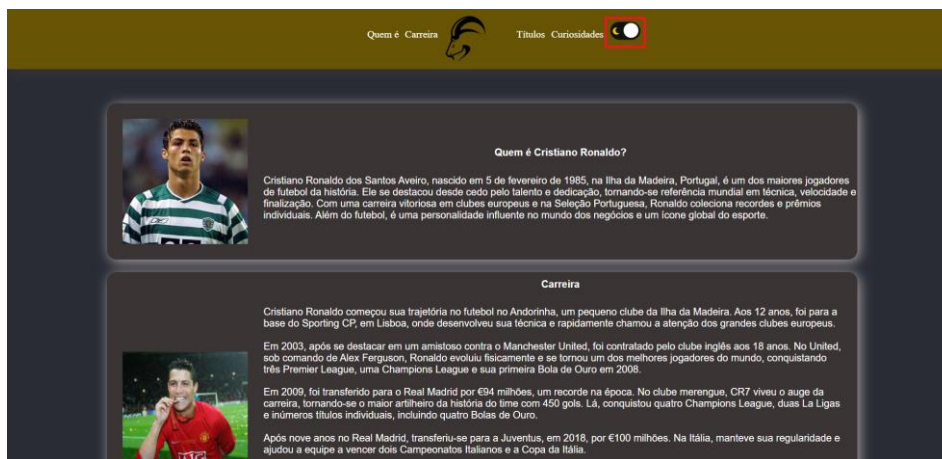
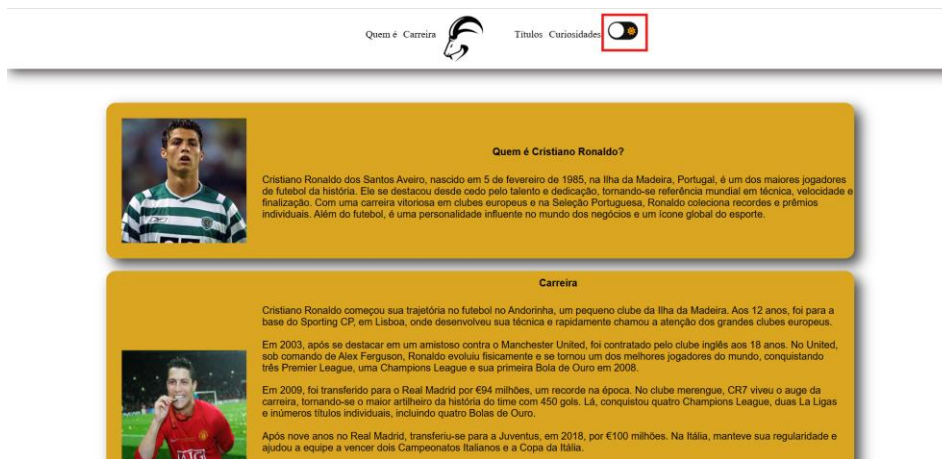
Por fim, no css, foi adicionada cores para a lua e o sol e o transform:translate significa que a bolinha irá se mexer por 24px quando o checkbox for checado.

Após isso, foi feito um código bem simples e funcional em javascript para fazer o efeito de trocar o tema

```
16
17 // Botão tema dark
18 const chk = document.getElementById('chk')
19
20 chk.addEventListener('change', () => {
21   document.body.classList.toggle('dark')
22 })
23
```

Foi criada uma constante que lê o Id de nome “chk” no documento, e é criado um evento, onde quando é marcado ou desmarcado (efeito de mudança), é executado o código que adiciona ou remove a classe dark do body.

Visualmente:



5ª Parte

Na 5ª Parte do projeto, inserimos uma 3ª funcionalidade em JavaScript, um botão que volta para o início da página que aparece somente quando a página chega ao fim.

Para implementar na página inserimos um botão no HTML com o id “btnTop”:

```
<button id="btnTop" title="Voltar ao topo">⬆️</button>
```

A partir do id btnTop modificamos o campo para aplicar a função em JavaScript.

```
// Botão voltar ao topo
const btnTop = document.getElementById("btnTop");

window.onscroll = function () {
  if (document.body.scrollTop > 300 || document.documentElement.scrollTop > 300) {
    btnTop.style.display = "block";
  } else {
    btnTop.style.display = "none";
  }
};

btnTop.addEventListener("click", function () {
  window.scrollTo({
    top: 0,
    behavior: "smooth"
  });
});
```

Foi criada uma constante que lê o ID "btnTop" no documento, e é criado um evento que detecta a rolagem da página (onscroll). Quando a rolagem ultrapassa 300 pixels, o botão é exibido na tela, caso contrário, ele é escondido. Também é criado um evento de clique nesse botão, que ao ser acionado faz a página rolar suavemente até o topo (efeito de rolagem suave).

Visualmente:

