

Relatório do meu trabalho final de programação web

Eder Lopes Pires



Formando- Eder Lopes Pires

Formador- Willian Pires

Introdução

Este trabalho final de programação web visa criar um portfólio pessoal simples e eficaz. O objetivo é desenvolver uma plataforma que não apenas apresente projetos e realizações de forma clara, mas também seja fácil de navegar e visualmente atraente.

Utilizando tecnologias como HTML, CSS e possivelmente JavaScript, o projeto focará em criar um layout responsivo que se adapte a diferentes dispositivos. A simplicidade será chave, garantindo uma experiência de usuário intuitiva sem excesso de informações desnecessárias.

Além disso, o trabalho considerará aspectos básicos de acessibilidade e otimização para mecanismos de busca, assegurando que o portfólio seja acessível a todos os usuários e fácil de encontrar online.

Ao final, o objetivo é não apenas demonstrar habilidades de desenvolvimento web, mas também oferecer uma ferramenta prática e eficaz para promover o perfil profissional em um ambiente digital cada vez mais competitivo.

Desenvolvimento

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Meu Portefólio</title>
7   <link rel="stylesheet" href="estilo.css">
8   <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/font-awesome@5.15.4/css/all.min.css">
9 </head>
10 <body>
11   <header>
12     <nav>
13       
14       <ul>
15         <li><a href="#"><span id="cores">Eu</span></a></li>
16         <li><a href="#"><span id="cores">Serviços</span></a></li>
17         <li><a href="#"><span id="cores">Contactos</span></a></li>
18       </ul>
19     </nav>
20   </header>
```

O código começa com uma marcação que indica o fim da seção `<head>` e o início do `<body>`, onde o conteúdo visível para o usuário é colocado.

`<header>`: Define a área de cabeçalho da página, que geralmente contém elementos como logotipo, menu de navegação, etc.

`<nav>`: Define uma seção de navegação, geralmente contendo links para diferentes partes do site.

``: Exibe um logotipo na barra de navegação. O atributo `src` especifica o caminho para a imagem do logotipo e o atributo `alt` fornece um texto alternativo para acessibilidade.

`` e ``: Utilizados para criar uma lista de itens de navegação.

- `` (unordered list): Contém os itens da lista.
- `` (list item): Cada item da lista.
- ``: Define um link. O atributo `href="#"` cria um link vazio que pode ser posteriormente preenchido com URLs reais.

:

`id="cores"`: Define um identificador (ID) chamado "cores". IDs devem ser únicos dentro de um documento HTML, mas neste caso, o mesmo ID está sendo usado repetidamente. Idealmente, classes (`class="cores"`) seriam mais apropriadas para estilização em CSS.

```
1  body{
2      background: #080808; ;
3      color: #fff;
4
5  }
6  header{
7
8      background-size: cover;
9      background-position: center;
10     margin: 50px
11
12 }
13
14
15
16
17
18 #image{
19     height :700px;
20     width : 700px;
21     border-radius: 20px;
22
23
24 }
25 .container{
26     padding:10px 10px;
27 }
28 nav{
29     display: flex;
30     align-items: center;
31     justify-content: space-between;
32     flex-wrap: wrap;
33     margin: 50px;
34
35 }
36
37 nav ul li{
38     display: inline-block;
39     list-style: none;
40     margin: 10px 20px;
41
42
43 }
44 nav ul li a{
45     color: #fff;
46     text-decoration: none;
47     font-size: 25px;
48     position: relative;
```

Estilos do Corpo (body):

- `background: #080808;` Define o fundo do corpo da página com a cor hexadecimal #080808, que representa um tom de cinza escuro.
- `color: #fff;` Define a cor do texto dentro do corpo da página como branco (#fff), garantindo bom contraste com o fundo escuro.

Estilos do Cabeçalho (header):

- `background-size: cover;` Garante que a imagem de fundo do cabeçalho cubra toda a área disponível sem distorção.
- `background-position: center;` Centraliza a imagem de fundo do cabeçalho.
- `margin: 50px;` Define uma margem de 50 pixels ao redor do cabeçalho, ajudando a separá-lo do conteúdo adjacente.

Estilos da Imagem (#imagem):

- `height: 700px;` Define a altura da imagem como 700 pixels.
- `width: 700px;` Define a largura da imagem como 700 pixels.
- `border-radius: 20px;` Aplica um raio de 20 pixels às bordas da imagem, criando cantos arredondados.

Estilos do Container (container):

- `padding: 10px 10px;` Define um preenchimento interno de 10 pixels em todas as direções (topo, direita, baixo, esquerda) para o container.

Estilos da Navegação (nav, nav ul li, nav ul li a):

- `nav:` Define a navegação como um layout flexível com itens alinhados vertical e horizontalmente.
- `nav ul li:` Estiliza os itens da lista de navegação como blocos em linha, sem marcadores de lista e com margens específicas.
- `nav ul li a:` Estiliza os links dentro dos itens da lista de navegação com cor branca, sem sublinhado, tamanho de fonte de 25 pixels e posicionamento relativo.

```

22 <div class="section1">
23
24     <h1 class="outros"> Developer <span id="cores">Software</span>
25         <br>
26
27         Olá, sou o <span id="nova">Eder Pires</span>
28         <br>
29         Sou desenvolvedor de <span id="cores">Software</span>
30         <br>
31         Desenvolvedor de <span id="cores">Jogos</span>
32         <br>
33         Sou Designer de paginas <span id="cores">Web</span>
34     <br>
35 </h1>
36 
37 </div>
38

```

□ Estrutura Geral:

- `<div class="section1">`: Define uma seção ou divisão do conteúdo, identificada pela classe `section1`.

□ Conteúdo dentro de `section1`:

- `<h1 class="outros"> Developer Software ... </h1>`: Um título de nível 1 (`<h1>`) com a classe `outros`. Contém vários spans com IDs e texto dentro. Aqui está uma análise dos spans:
 1. ``: Utilizado várias vezes para definir partes do texto com a mesma cor. IDs devem ser únicos dentro de um documento



`display: flex;`: Define um layout flexível para os elementos dentro da classe `.section1`, permitindo um controle fácil sobre o alinhamento e o espaçamento dos elementos filhos.

`justify-content: space-between;`: Distribui os elementos filhos horizontalmente, com espaço entre eles. Isso significa que o espaço disponível será dividido de modo que haja espaço igual entre os elementos, colocando-os nos extremos do contêiner `.section1`.

`margin: 50px;`: Define uma margem externa de 50 pixels ao redor de `.section1`, criando um espaçamento uniforme entre esta seção e os elementos circundantes.

Css

1.

```
51  .section1{
52      display: flex;
53      justify-content: space-between;
54      margin: 50px;
55
56
57
58  }
59  .outros{
60      margin-top: 300px;
61  }
62  #cores{
63      color:  rgb(223, 15, 15);
64  }
65  #nova{
66      color:  rgb(223, 15, 15);
67  }
```

Estilos da .section1

- display: flex;: Define um layout flexível para os elementos dentro da classe .section1, permitindo um controle fácil sobre o alinhamento e o espaçamento dos elementos filhos.
- justify-content: space-between;: Distribui os elementos filhos horizontalmente, com espaço entre eles. Isso significa que o espaço disponível será dividido de modo que haja espaço igual entre os elementos, colocando-os nos extremos do contêiner .section1.
- margin: 50px;: Define uma margem externa de 50 pixels ao redor de .section1, criando um espaçamento uniforme entre esta seção e os elementos circundantes.

CSS

- HTML, mas neste caso, o mesmo ID está sendo usado repetidamente. Seria mais apropriado usar classes (class="cores") para estilização em CSS.
- : Define um span com ID nova, usado uma vez para destacar o nome "Eder Pires".
- : Uma imagem com o ID imagem e um caminho (src) para o arquivo "tercer-removebg-preview.png". O atributo alt está vazio, seria ideal fornecer um texto alternativo descritivo para acessibilidade.

Estilos da outros

- margin-top: 300px;: Define uma margem superior de 300 pixels para o elemento com a classe .outros. Isso empurra o elemento para baixo, criando um espaço em relação aos elementos acima dele na página.

Estilos dos IDs (#cores, #nova, #logo):

- #cores e #nova: Ambos definem a cor do texto como vermelho utilizando o valor RGB rgb(223, 15, 15). Isso garante que o texto dentro dos elementos com esses IDs seja exibido em vermelho.
- #logo: Define o tamanho da fonte para o elemento com ID logo como 25 pixels, possivelmente para destacar um elemento específico na página com um tamanho de texto maior.

```
77     }
78     .lista-de-Serviços{
79         display: grid;
80         grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));
81         grid-gap: 40px;
82         display: flex;
83         flex-direction: row;
84         margin-left: 150px;
85     }
86     .lista-de-Serviços div{
87         justify-content: center;
88         margin-left: 130px;
89         margin-top: 170px;
90     }
91
92
93     image{
94         height :600px;
95         width : 600px;
96     }
```

- **Estrutura Geral:**

- <div class="lista-de-Serviços">: Define uma seção ou divisão do conteúdo, identificada pela classe lista-de-Serviços.
- : Uma imagem com o ID imagem e um caminho (src) para o arquivo "informatica.jpg". O atributo alt está vazio, seria ideal fornecer um texto alternativo descritivo para acessibilidade.

- **Conteúdo dentro de .lista-de-Serviços:**

- `<h1>Meus Serviços</h1>`: Um título de nível 1 (`<h1>`) com o texto "Meus Serviços", onde o texto dentro do span com ID `cores` provavelmente será estilizado com uma cor específica.
- Vários `<h2>` e `<p>` dentro do `<div>`: Esses elementos definem diferentes serviços oferecidos, como Web Designer, programador Full-Stack, programador de Jogos e programador de aplicativos Desktop. Cada um deles usa spans com ID `cores` para estilização do texto em vermelho.

```

38
39 <div class =".lista-de-Serviços">
40   
41   <div>
42     <h1><span id="cores">Meus Serviços</span></h1>
43     <br>
44     <h2><span id="cores">Web Designer</span></h2>
45     <br>
46     <p>Trabalho com <span id="cores">HTML, CSS, FIGMA</span></p>
47     <br>
48     <h2>programador <span id="cores">Full-Stack</span></h2>
49     <br>
50     <p>Trabalho com programação <span id="cores">Front-End e Back-End</span></p>
51     <br>
52     <h2>programador de <span id="cores">Jogos</span></h2>
53     <br>
54     <p>Trabalho com aplicativos como <span id="cores">Java, python, Java-Script</span></p>
55     <br>
56     <h2>programador de aplicativos <span id="cores">Desktop</span></h2>
57     <br>
58     <p>Trabalho com aplicativos como <span id="cores">Java, python, Java-Scrip</span></p>
59   </div>
60 </div>

```

.lista-de-Serviços:

- `display: grid;`: Define um layout em grade para os elementos dentro de `.lista-de-Serviços`, permitindo uma organização eficiente dos itens.
- `grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));`: Define as colunas da grade. `auto-fit` ajusta automaticamente o número de colunas com base no tamanho do conteúdo, variando entre 250 pixels (largura mínima) e 1fr (fração do espaço disponível). Isso permite que os itens se ajustem ao tamanho da tela de forma responsiva.
- `grid-gap: 40px;`: Define o espaçamento entre as células da grade como 40 pixels, criando um espaçamento visual entre os itens.
- `display: flex;`: Aplicação redundante de `display: flex;` na classe `.lista-de-Serviços`. No entanto, isso não tem efeito adicional, pois a propriedade `display` só pode ser definida uma vez para um elemento.
- `flex-direction: row;`: Define a direção dos itens flexíveis como horizontal (linha), o que é redundante dado o uso de `grid` anteriormente.
- `margin-left: 150px;`: Define uma margem esquerda de 150 pixels aplicada à `.lista-de-Serviços`, posicionando-a na página.

.lista-de-Serviços div:

- justify-content: center;; Centraliza o conteúdo horizontalmente dentro dos divs que estão dentro de .lista-de-Serviços, o que pode ser útil para alinhar o conteúdo dentro das células da grade.
- margin-left: 130px;; Define uma margem esquerda de 130 pixels aplicada aos divs dentro de .lista-de-Serviços, criando um espaçamento adicional à esquerda dos itens dentro da grade.
- margin-top: 170px;; Define uma margem superior de 170 pixels aplicada aos divs dentro de .lista-de-Serviços, posicionando-os mais abaixo na página.

image:

- height: 600px;; Define a altura da imagem como 600 pixels, o que pode ser útil para padronizar o tamanho das imagens exibidas.
- width: 600px;; Define a largura da imagem como 600 pixels, garantindo consistência no tamanho das imagens exibidas.

```
68
69     <div class="Formulario">
70
71         <form action="">
72             <h1>Login</h1>
73             <div class="input-box">
74                 <input type="text"placeholder = "Username" required>
75                 <i class="fa-solid fa-user"></i>
76                 <div class="input-box">
77                     <input type="password"placeholder = "Password" required>
78                     <i class="fa-solid fa-lock"></i>
79                 </div>
80                 <div class="Remember-forgot">
81                     <label><input type="checkbox">Remember me</label>
82                     <a href="#">Forgot password</a>
83                 </div>
84                 <button type="Submit" class="btn">Login</button>
85                 <div class="Register-link">
86                     <p>Don't have an account?</p>
87                     <a href="#">Register</a>
88                 </div>
89             </form>
90     </div>
91
92     <footer id="footer">
```

Estrutura Geral:

- `<div class="Formulario">`: Define uma seção ou divisão do conteúdo com a classe Formulario.
- `<form action="">`: Define um formulário HTML que, quando enviado, envia os dados para o mesmo URL (`action=""` indica que o formulário enviará para o próprio URL).
- `<h1>Login</h1>`: Título do formulário, indicando que este é um formulário de login.

Elementos do Formulário:

- `<div class="input-box">`: Contém campos de entrada (`<input>`) para o nome de usuário e senha, juntamente com ícones representados por classes de ícone (`fa-solid fa-user` e `fa-solid fa-lock`).
 - `<input type="text" placeholder="Username" required>`: Campo de entrada para o nome de usuário, obrigatório (required).
 - `<i class="fa-solid fa-user"></i>`: Ícone de usuário para o campo de nome de usuário.
- `<div class="input-box">`: Outro conjunto de campos de entrada para senha.
 - `<input type="password" placeholder="Password" required>`: Campo de entrada para senha, com tipo definido como password para ocultar o texto digitado, obrigatório (required).
 - `<i class="fa-solid fa-lock"></i>`: Ícone de cadeado para o campo de senha.
- `<div class="Remember-forgot">`: Contém opções de lembrar senha e link para recuperação de senha.
 - `<label><input type="checkbox">Remember me</label>`: Checkbox para lembrar o usuário (opção "Remember me").
 - `Forgot password`: Link para recuperação de senha.
- `<button type="submit" class="btn">Login</button>`: Botão de envio do formulário, com tipo definido como submit para enviar os dados inseridos no formulário.
- `<div class="Register-link">`: Contém um link para o registro de uma nova conta.
 - `<p>Don't have an account?</p>`: Texto indicando que não há uma conta registrada.
 - `Register`: Link para a página de registro.

```
231
232 .register-link {
233     font-size: 14px;
234     text-align: center;
235     margin-top: 20px;
236 }
237
238 .register-link p {
239     margin-bottom: 0;
240 }
241
242 .register-link a {
243     color: #fff;
244     text-decoration: none;
245     font-weight: 600;
246 }
247
248 .register-link a:hover {
249     text-decoration: underline;
250 }
251
252
253
254 #video{
255     display: flex;
256     justify-content: center;
257     align-items: center;
258 }
259 #footer{
260     display: flex;
261     flex-direction: column;
262 }
```

```
187 .input-box i {
188     position: absolute;
189     right: 10px;
190     top: 50%;
191     transform: translateY(-50%);
192     color: ■ #fff;
193 }
194
195 .remember-forgot {
196     display: flex;
197     justify-content: space-between;
198     font-size: 14px;
199     margin-bottom: 15px;
200 }
201
202 .remember-forgot label {
203     display: flex;
204     align-items: center;
205     color: ■ #fff;
206 }
207
208 .remember-forgot a {
209     color: ■ #fff;
210     text-decoration: none;
211 }
212
213 .remember-forgot a:hover {
214     text-decoration: underline;
215 }
216
217 .btn {
218     width: 100%;
219     height: 45px;
220     background: ■ #fff;
221     border: none;
222     outline: none;
223     border-radius: 40px;
224     box-shadow: 0 0 10px □ rgba(0, 0, 0, 0.5);
225     cursor: pointer;
226     font-size: 16px;
227     color: □ #333;
228     font-weight: 600;
229     margin-top: 10px;
230 }
231
```

```
142 .Formulario {
143     display: flex;
144     justify-content: center;
145     align-items: center;
146     gap: 100px;
147
148
149 }
150
151 form {
152     width: 320px;
153     background-color: #555;
154     color: #fff;
155     padding: 30px;
156     border-radius: 10px;
157     box-shadow: 0 0 10px rgba(0, 0, 0, 0.5);
158 }
159
160 form h1 {
161     font-size: 24px;
162     text-align: center;
163     margin-bottom: 20px;
164 }
165
166 .input-box {
167     width: 100%;
168     position: relative;
169     margin-bottom: 20px;
170 }
171
172 .input-box input {
173     width: 100%;
174     height: 40px;
175     padding: 10px;
176     background: transparent;
177     border: 2px solid #fff;
178     border-radius: 20px;
179     color: #fff;
180     font-size: 16px;
181 }
182
183 .input-box input::placeholder {
184     color: #ccc;
185 }
```

.Formulario:

- display: flex;; Define um layout flexível para centralizar o formulário na página, tanto horizontalmente (justify-content: center) quanto verticalmente (align-items: center).
- gap: 100px;; Define um espaçamento de 100 pixels entre os elementos filhos dentro da .Formulario.

form:

- width: 320px;; Define a largura fixa do formulário como 320 pixels.
- background-color: #555;; Define a cor de fundo do formulário como cinza escuro (#555).
- color: #fff;; Define a cor do texto dentro do formulário como branco.
- padding: 30px;; Define um espaçamento interno de 30 pixels dentro do formulário.
- border-radius: 10px;; Arredonda as bordas do formulário com um raio de 10 pixels.
- box-shadow: 0 0 10px rgba(0, 0, 0, 0.5);; Adiciona uma sombra suave ao redor do formulário para efeito de profundidade.

.input-box:

- width: 100%;; Define a largura total do contêiner de entrada.
- position: relative;; Define uma posição relativa para os elementos filhos absolutamente posicionados.
- margin-bottom: 20px;; Define um espaçamento inferior de 20 pixels entre os elementos .input-box.

.input-box input:

- width: 100%;; Define a largura total do campo de entrada.
- height: 40px;; Define a altura do campo de entrada como 40 pixels.
- padding: 10px;; Adiciona um preenchimento interno de 10 pixels ao campo de entrada.
- background: transparent;; Define um fundo transparente para o campo de entrada.
- border: 2px solid #fff;; Adiciona uma borda sólida branca ao redor do campo de entrada.
- border-radius: 20px;; Arredonda as bordas do campo de entrada com um raio de 20 pixels.
- color: #fff;; Define a cor do texto dentro do campo de entrada como branco.
- font-size: 16px;; Define o tamanho da fonte do texto dentro do campo de entrada como 16 pixels.

.input-box input::placeholder:

- color: #ccc;; Define a cor do texto de placeholder (texto temporário dentro do campo de entrada) como cinza claro (#ccc).

.input-box i:

- position: absolute;; Define uma posição absoluta para os ícones dentro do campo de entrada.
- right: 10px;; Alinha os ícones à direita com um deslocamento de 10 pixels da borda direita do campo de entrada.
- top: 50%;; Alinha os ícones verticalmente ao centro.
- transform: translateY(-50%);; Ajusta a posição vertical dos ícones para centralizá-los completamente verticalmente.
- color: #fff;; Define a cor dos ícones como branco.

.remember-forgot:

- display: flex;; Define um layout flexível para os elementos "Lembrar-me" e "Esqueceu a senha".
- justify-content: space-between;; Distribui os elementos horizontalmente, criando espaço entre eles.
- font-size: 14px;; Define o tamanho da fonte para os elementos dentro de .remember-forgot.
- margin-bottom: 15px;; Define um espaçamento inferior de 15 pixels para .remember-forgot.

.remember-forgot label:

- display: flex;; Define um layout flexível para o rótulo "Lembrar-me".
- align-items: center;; Alinha os itens verticalmente dentro do rótulo.


```

3      92  <footer id="footer">
      93
      94      <div id="Contacto">
18.5... 95          <div class="Container">
      96
      97              <div class="Outros Contacto">
      98                  <h1 class="Sub-titulo">Contactar-me</h1>
      99                  <p>238 584 32 89</p>
100                  <div class="Icons Socias">
101                      <a href="#"><i class="fab fa-facebook-f"></i></a>
102                      <a href="#"><i class="fab fa-instagram"></i></a>
103                      <a href="#"><i class="fab fa-facebook-messenger"></i></a>
104                      <a href="#"><i class="fab fa-whatsapp"></i></a>
105                  </div>
106              </div>
107          </div>
108      </div>
109  </footer>
110 </body>
111 </html>

```

Estrutura Geral:

- `<footer id="footer">`: Define o rodapé da página com a identificação footer.
- `<div id="Contacto">`: Uma divisão que contém informações de contato.
- `<div class="Container">`: Define um contêiner para organizar o conteúdo dentro do rodapé.
- `<div class="Outros Contacto">`: Uma seção específica para informações de contato adicionais.
- `<h1 class="Sub-titulo">Contactar-me</h1>`: Título secundário que indica "Contactar-me".
- `<p>238 584 32 89</p>`: Um parágrafo com um número de telefone.
- `<div class="Icons Socias">`: Contém ícones de redes sociais para contato.
- Ícones de redes sociais dentro de `<a>` tags usando classes do FontAwesome (por exemplo, `fab fa-facebook-f`, `fab fa-instagram`, etc.).

Conclusão

Com muita esforço e empenho conclui o meu trabalho final de programação web sobre o meu portefolio pessoal. Tive baste dificuldade em termina-lo pois não tenho computador pessoal e nem acesso a internet mas mesmo assim esforcei bastante na sala de aula, sacrifiquei varias aulas para conseguir o meu trabalho. Ficou um pouco incompleto mas foi o meu maximo que consegui fazer. Mas apesar de tudo adquiri varios conseitos base em relação a desenvolvimento de página web. Em termos de semantica, estilização e funções em javaScrip. É uma base que vou levar para vida toda pois adquiri bastante conhecimento e principios sobre desenvolvimento de paginas web.