

Aula 14

Sites Single Page

O que é um Site Single Page

Um site Single Page é aquele que possui apenas uma página, seus links não levam para páginas secundárias, apenas descem a mesma página para exibir seu conteúdo mais abaixo. Contudo, não é todo tipo de conteúdo que pode ser construído como Single Page.

Sites de empresas maiores ou que oferecem muitos produtos, não se encaixam bem para o modelo Single Page, pois o conteúdo poderá ficar incompleto ou confuso em uma página só. Por isso, é recomendado utilizar este modelo para pequenas empresas, como página de apresentação ou para eventos, que possuem informações mais pontuais.

Criando um cartão de visitas virtual

Sites Single Page não possuem um formato ou elementos específicos usados, mas são marcantes por utilizarem design moderno e investirem em efeitos e animações. Por isso, nosso site desta aula será um cartão de visitas digital com animações.

Para iniciar, crie a pasta da aula e adicione um arquivo `index.html`, outro `style.css` e uma pasta `img`.

HTML

A estrutura inicial do HTML terá 6 Divs. A primeira, que engloba todas as outras, terá a classe `"frame"`. A próxima, fica dentro da `"frame"` e tem a classe `"card"`, que é a limitação do nosso cartão. Dentro da `"card"` temos outra div chamada `"flip"`, que será responsável por virar o cartão para ler a face de trás. E por fim, dentro da `"flip"` há 2 divs com mesma hierarquia: uma com classe `"front"` e a outra com classe `"back"`. Dentro da `"back"` ainda há uma última div, sem classe.

```

1  <!DOCTYPE html>
2  <html lang="pt-BR" >
3
4  <head>
5      <meta charset="UTF-8">
6      <title></title>
7      <link rel="stylesheet" href="style.css">
8  </head>
9
10 <body>
11
12     <div class="frame">
13         <div class="card">
14             <div class="flip">
15
16                 <div class="front">
17
18                 </div>
19
20                 <div class="back">
21                     <div>
22                     </div>
23                 </div>
24             </div>
25         </div>
26     </div>
27
28 </body>
29 </html>

```

Vamos começar a preencher estas divs então: na div “front”, vamos adicionar um título h1 e um parágrafo com classe “flip-me”. Essa é a face frontal do nosso cartão, é o que aparecerá inicialmente ao carregar a página.

```

<div class="front">
    <h1>hey!</h1>
    <p class="flip-me">Flip Me</p>
</div>

```

Na div seguinte, “back”, adicione um título h3 para o seu nome (pode ser o nome completo) e um título h2 para seu título profissional. Na div interna, é onde serão alocados os ícones para redes externas, como Github, LinkedIn, etc, pode adicionar as redes que preferir anexar.

As imagens para os links externos, podem ser baixadas no site [Falticon](#), há diversos modelos com designs diferentes. Não esqueça de colocar as imagens salvas na pasta img e de nomeá-las corretamente no HTML.

```
<div class="back">
  <h3>Tauana</h3>
  <h2>Web Developer</h2>
  <div>
    <a href=""></a>
    <a href=""></a>
    <a href=""></a>
  </div>
</div>
```

CSS

Antes de começarmos a configurar nosso CSS, precisamos importar a fonte de texto Nunito, do Google Fonts. Desta forma podemos determinar que todo o body seguirá esta Família de Fonte.

```
@import url("https://fonts.googleapis.com/css?family=Nunito|Satisfy");

body {
  background: #643a7a;
  font-family: "nunito", sans-serif;
}
```

O seletor seguinte, é para a div com classe "frame". Essa div cobrirá o fundo do cartão, formando uma moldura ao redor, também ajudará com que o cartão fique na posição correta. Por isso, usamos posição absoluta e ajustamos o topo, tamanho e margens.

```
.frame {  
  position: absolute;  
  background-color: #201c29;  
  top: 50%;  
  left: 50%;  
  width: 400px;  
  height: 400px;  
  margin-top: -200px;  
  margin-left: -200px;  
  box-shadow: 8px 16px 16px rgba(0, 0, 0, 0.6);  
}
```

```
.frame, .card {  
  display: flex;  
  align-items: center;  
  justify-content: center;  
}
```

A classe “flip-me” pertence ao parágrafo com os mesmos dizeres, é apenas um recurso visual. Uma propriedade interessante que ainda não havíamos usado é a “transform” que pode transformar um elemento html de acordo com o valor atribuído. Nesse caso utilizamos rotate, para rotacionar em um ângulo de -90°.

```
.flip-me {  
  position: absolute;  
  top: 30%;  
  left: 77%;  
  font-size: 1.3rem;  
  transform: rotate(-90deg);  
}
```

Vamos determinar algumas questões visuais para os textos:

```
.front h1 {  
  font-family: Satisfy, cursive;  
  font-size: 128px;  
  margin: 16px 80px;  
  color: #ef2f96;  
}
```

```
p {  
  position: absolute;  
  top: 60%;  
  left: 10%;  
  color: rgba(255, 255, 255, 0.2);  
}
```

A classe “flip”, como citado anteriormente, será a responsável por de fato fazer o cartão virar. Quando passamos o mouse sobre o cartão, possui uma propriedade “transform” que rotaciona a div no eixo X e realiza uma translação 3D.

```
.card:hover .flip {  
  transform: rotateX(180deg) translate3d(0, 0, 0);  
  box-shadow: 8px -10px 15px 0 rgba(0, 0, 0, 0.5);  
}  
  
.flip {  
  transform-style: preserve-3d;  
  transition: all 0.7s ease-in-out;  
  perspective: 1000px;  
}
```

Até então, a face da frente e das costas do cartão estavam se sobrepondo e aparecendo juntas na tela, por isso o próximo passo é fazer com que apenas uma delas apareça por vez.

Utilizamos uma propriedade nova chamada “webkit”, que é uma extensão CSS utilizada pelo Safari e Chrome. Neste caso, utilizamos “-webkit-backface-visibility”, que define quando uma face deve ficar escondida (hidden) ou não.

```

.front, .back {
  -webkit-backface-visibility: hidden;
  width: 100%;
  height: 100%;
  background: #201c29;
}

.card .back {
  transform: rotateX(-180deg);
  position: absolute;
  top: 0;
  left: 0;
}

```

Vamos ajustar também o alinhamento e fontes dos textos da face de trás.

```

.back {
  letter-spacing: 2px;
  text-align: center;
}

.back h2 {
  font-size: 36px;
  margin: 8px;
  color: #ef2f96;
}

.back h3 {
  color: rgba(255, 255, 255, 0.7);
  font-family: satisfy, cursive;
  font-weight: 300;
  font-size: 32px;
  margin: 32px;
}

```

E por fim, ajustamos as imagens, para que tenham o tamanho correto. Também adicionamos um hover, para que aumentem o tamanho ao passar o mouse.

```
img{  
  width: 28px;  
}  
  
img:hover {  
  width: 48px;  
  cursor: pointer;  
}
```

Tarefa de Casa

Personalize seu cartão de visitas adicionando informações relevantes, como número de telefone, e-mail, outras redes, uma frase que te descreva, etc. As informações adicionadas vão do critério de cada um, você pode também modificar as cores e fontes para deixá-lo ao seu gosto.