

Universidade Paulista

Ciência da computação

Relatório referente a atividade solicitada para entrega até o dia
08/05

DESENVOLVIMENTO DO SITE RESPONSIVO BELFEST

Diogo Sousa Nogueira - N539CF-0

Enrico Brosque Palma - F271EI-0

Bauru
2020-05-08

Introdução

Relatório sobre o desenvolvimento do website Belfast (Nome ilustrativo). O website consiste na apresentação de um modelo de automóvel onde o cliente pode rotacionar em vários ângulos o automóvel, para melhor visualização, escolher cor de lataria, modelos de calotas produzidos em fábrica de acordo com sua montadora original (BMW no caso) e detalhes como faixas e cores de maçanetas.

No trabalho foi utilizado algumas tecnologias para o desenvolvimento do website, dentre elas:

- Figma
- Visual Studio Code

linguagens de marcação:

- HTML5
- Css3;

Linguagem de programação:

- JavaScript;

Framework:

- x3DOM
- Bootstrap 4.4.1

Uso do Figma

Protótipos

Foram feitos três diferentes protótipos até chegarmos no design preferido.

Tínhamos como ideia base o automóvel e sua paleta de cores centralizados na tela a todo momento, um fundo preto/cinza e uma faixa branca atrás do carro para adicionar contraste.



Protótipo Final

Uso do HTML5/CSS3/Bootstrap 4

Navbar

Projetamos o site com a responsividade oferecida pelo Bootstrap em mente. Foi escolhido a barra de navegação visando um site completo, onde o usuário tem a opção de navegar com facilidade por diversos conteúdos que a página pode oferecer (no caso, utilizamos apenas como placeholders).

Graças a framework e suas classes pré-definidas nós adicionamos uma barra de navegação (Navbar) que dependendo da resolução, se adapta em um menu de hamburger.

```
<!--Navbar-->
<div>
  <nav class="navbar navbar-expand-md navbar-dark bg-dark">
    <a href="store.html" class="navbar-brand">
      
    </a>
  <!--Hamburger-->
    <button class="navbar-toggler" data-toggle="collapse" data-target="#navbarMenu">
      <span class="navbar-toggler-icon"></span>
    </button>
  <!--//Hamburger-->
  <!--Nav Elements-->
    <div class="collapse navbar-collapse" id="navbarMenu">
      <ul class="navbar-nav">
        <li class="nav-item">
          <a href="#" class="nav-link">Selecione um Veículo</a>
        </li>
        <li class="nav-item">
          <a href="#" class="nav-link">Encontre um Vendedor</a>
        </li>
        <li class="nav-item">
          <a href="#" class="nav-link">Montagem e Preço</a>
        </li>
      </ul>
    </div>
  <!--//Nav Elements-->
  </nav>
</div>
<!--//Navbar-->
```

Uso do X3DOM/CSS3/JavaScript

Conteúdo

No software Visual Studio Code foi onde se passou um dos maiores momentos de desenvolvimento, a ideia inicial se passava por volta do desenvolvimento de um website com o foco em manter um automóvel como estrela principal do site, assim a ideia de mantê-lo sempre no centro de tudo com um grande destaque, levando em consideração esta ideia o primeiro passo foi fazer com que o viewpoint tivesse seu tamanho alterado e trazer ele para o centro.

X3d é uma tag da framework x3DOM onde se localiza o quadro 3d que estava o carro, esta tag traz nela algumas características que a definem com o modelo padrão que a própria framework traz, então o necessário a ser feito era alterar estas configuração para trazer o resultado desejado.

```
<x3d id='x3d' showStat='false' showLog='false' height='500px' style="display:inline;">
```

Nesta linha podemos ver o ID 'x3d', foi neste id que usei para fazer as alterações do contêiner do automóvel 3d, esta linha foi escolhida pois por ordem de hierarquia logo após dela se concentra todo o conteúdo do automóvel, assim após alguns testes foi descoberto que esta linha comandava o espaçamento de onde seria apresentado o modelo em 3D dentro do HTML. Foi determinado uma altura de 500px fixa tanto para telas de computadores como tela de celulares e demais dispositivos, pois através de testes de resolução esta seria uma das melhores opções para manter a qualidade.

Logo em seguida foi feita as alterações dentro da ID 'x3d', está ID se encontrava no arquivo de estilo CSS do framework X3DOM, fiz um acesso ao arquivo e realizei algumas alterações para alcançar o objetivo.

```
X3D, x3d {
  position:relative;
  cursor:pointer;
  margin: 0;
  padding: 0 !important;
  outline: none;
  display: block;
}

X3D:hover,
x3d:hover,
.x3dom-canvas:hover {
  -webkit-user-select: none;
  -webkit-touch-callout: none;
}

.x3dom-canvas {
  cursor:pointer;
  cursor:-webkit-grab;
  cursor:grab;
  width:100%;
  height:500px;
  right: 100%;
  border: 1px solid #fff;
}
```

Na linha x3d foi feita as seguintes alterações:

Alteração na propriedade de ID Position de absolute relative para que o quadro 3d pode-se ser alterado e que considerasse os demais itens que existem a sua volta;

Alteração na propriedade de ID Margin e padding ambos foram zerados para não haver espaçamentos que atrapalhassem a organização dos containers, tanto interna como externa.

Alteração em classes também foram realizadas, a classe 'x3dom-canvas' foi alterada as seguintes propriedades de classe:

Width: 720px → 100%;

Height: 300px → 500px;

Border: gray → White;

As seguintes alterações foram feitas para que o automóvel se mantesse sempre no tamanho ideal de tela, mantendo uma responsividade melhor, assim mantendo sua largura sempre o total da resolução atual e sua altura a mesma do local de apresentação do automóvel, que se encontra na linha x3d com height='500px'.

Tal mudança só foi possível após uma pesquisa de como funcionava a montagem em HTML do gráfico 3D:

```
<div class="containerCarro" >
  <x3d id='x3d' showStat='false' showLog='false' height='500px' style="display:inline;">
    <scene DEF='scene'>
      <viewpoint orientation='-0.600069 0.799493 -0.0269947 0.77042' position='3.05702 2.90676 4.27241'></viewpoint>
```

X3d tag responsável pela criação da área interativa com o automóvel/gráfico 3d; Scene tag que faz uma chamada para renderizar a localização do carro; Viewpoint está alocado onde se localiza o carro dentro do plano 3D.

Com esta organização em mente, adicionei uma div com a seguinte classe 'containerCarro', esta div ficou responsável por criar o local de onde pacote inteiro do automóvel 3d irá aparecer e nele fizemos algumas alterações para ficar o mais próximo do protótipo.

```
.containerCarro{  
  height: 500px;  
  background-color: white;  
  transform: skewY(3deg);  
}
```


height → 500px; mantendo o tamanho idêntico ao tamanho de renderização do conteúdo 3D, para não haver bordas.

Mantendo o fundo do conteúdo em branco para ficar semelhante ao do viewpoint, assim parecendo ser uma coisa só.

Aplicado propriedade de classe transform 'skewY(3deg)' que é a responsável por rotacionar determinados eixos do contêiner, neste caso ficou responsável de rotacionar 3 graus o eixo Y para dar o efeito necessário e assim ficar o mais próximo do protótipo.



Modelo 3D aqui



Feito as seguintes alterações o conteúdo do site estava responsivo, estilizado de acordo com o prototipo proposto e podendo ser utilizado de computadores, celulares, tablets e os demais dispositivos.

Interação

Para existir a interação entre o visitante e as alterações do automóvel 3D, foi utilizado a linguagem de programação JavaScript, nela utilizamos alguns botões com o intuito de quando o visitante/cliente clicar, haver uma interação no código que aplica determinadas funções.

```
<div class="botao">
  <button onclick="trocaCor('black')" class="btn btn-warning">Preto Fosco</button>
  <button onclick="trocaCor('#a30000')" class="btn btn-warning">Vinho</button>
  <button onclick="trocaCor('#124dff')" class="btn btn-warning">Azul</button>
  <button onclick="trocaCor('#f58d42')" class="btn btn-warning">Dourado</button>
</div>
```

Foi aplicado uma Div para criar o espaço de onde existiria os botões, com o auxílio da classe 'botao'.

```
.botao{
  margin-top: 150px;
  padding: 10px;
  text-align: center;
  background-color: rgba(71, 71, 71, 0.699);
  transform: skewY(3deg);
}
```

A classe botão traz algumas propriedades da classe para personalizar o conteúdo e ficar o mais próximo possível do proposto no protótipo.

Margin-top de 150px, como a Div foi posicionada antes do conteúdo 3d do carro, esta propriedade vem com o

objetivo de criar um espaçamento de todo o conteúdo do website entre a barra de navegação e o conteúdo, sendo ele de 150px fixos.

A cor dos botões se manteve a mesma da barra de navegação por motivos estéticos e foi aplicado um transform skewY de 3 graus para encaixar junto com o conteúdo do automóvel.

Para aplicar a interação usamos JavaScript para criar uma função onde irá alterar uma das propriedades já incluídas dentro do código do automóvel 3d. Nesta função nos falamos que o termo que estiver dentro dos parênteses 'color'

```
function trocaCor(color){
  var cor = document.getElementById("body_color");
  cor.setAttribute("diffuseColor", color);
}
```

irá ser aplicado ao atributo diffuseColor que se encontra na linha que contém o Id "body_color". O termo a ser aplicado se encontra na linha dos botões, assim

que chamar a função ele faz a aplicação, e a interação entra em atividade assim que o usuário clicar sobre o botão, pois a função está sendo chamada em um evento 'Onclick'.

Para realizar a troca das calotas do automóvel, foi utilizado a mesma ideia do código acima, porém com o intuito de carregar novas imagens e não de trocar cores. Assim então foi necessário fazer a alteração no código para que fosse "setado" o atributo 'url' e não o atributo 'diffuseColor'. Desta maneira utilizamos um dataBase da própria

```
function trocaRoda(rodaEscolha){
  var roda = document.getElementById("rodas");
  roda.setAttribute("url", rodaEscolha);
}
```

Framework x3DOM, para fazer a troca das calotas, por modelos alternativos, logo os botões para esta troca, ficaram com pequenas alterações, onde o termo a ser alterado, virou neste caso uma url que destina o material da calota no database do

framework x3DOM.

```
<div class="botao3">
  <button onclick="trocaRoda('https://x3dom.org/x3dom/example/texture/car/felge3.jpg')" class="btn btn-dark">Calota Original</button>
  <button onclick="trocaRoda('https://x3dom.org/x3dom/example/texture/car/felge1.jpg')" class="btn btn-dark">Calota street</button>
  <button onclick="trocaRoda('https://x3dom.org/x3dom/example/texture/car/felge4.jpg')" class="btn btn-dark">Calota racing</button>
  <button onclick="trocaRoda('https://x3dom.org/x3dom/example/texture/car/felge5.jpg')" class="btn btn-dark">Calota racing-2</button>
</div>
```

Footer

O plano inicial para o rodapé era ter três divisões:

- Contato:
 - Endereço fictício.
 - E-mail
 - Número de telefone.
- Suporte:
 - Suporte On-line.
 - Central de F.A.Q (Perguntas Frequentes).
- Menu de acesso rápido (reflexo da Navbar):
 - Logo.
 - Selecione um Veículo.
 - Encontre um Vendedor.
 - Montagem e Preço.

Foi feito uma Div junto ao uso da classe criada 'footer' e nela foi atribuída características para manter a mesma funcional e coerente com todo o conteúdo antes criado.

```
<!--Footer-->
<div class="footer">
  <div class="footer-copy">
    &copy; Belfast.com | Designed by <a href="#" target="_blank"></a>
  </div>
</div>
<!--//Footer-->
```

“Após inúmeras falhas tentando reproduzir o design desejado, reduzi o conteúdo para algo muito básico contendo apenas a trademark de design e uma assinatura com uma pequena easter egg para alívio cômico.” – Enrico

Para manter o footer funcional, responsivo e coerente com todo o conteúdo antes criado, foi feito algumas alterações no corpo todo através do CSS.

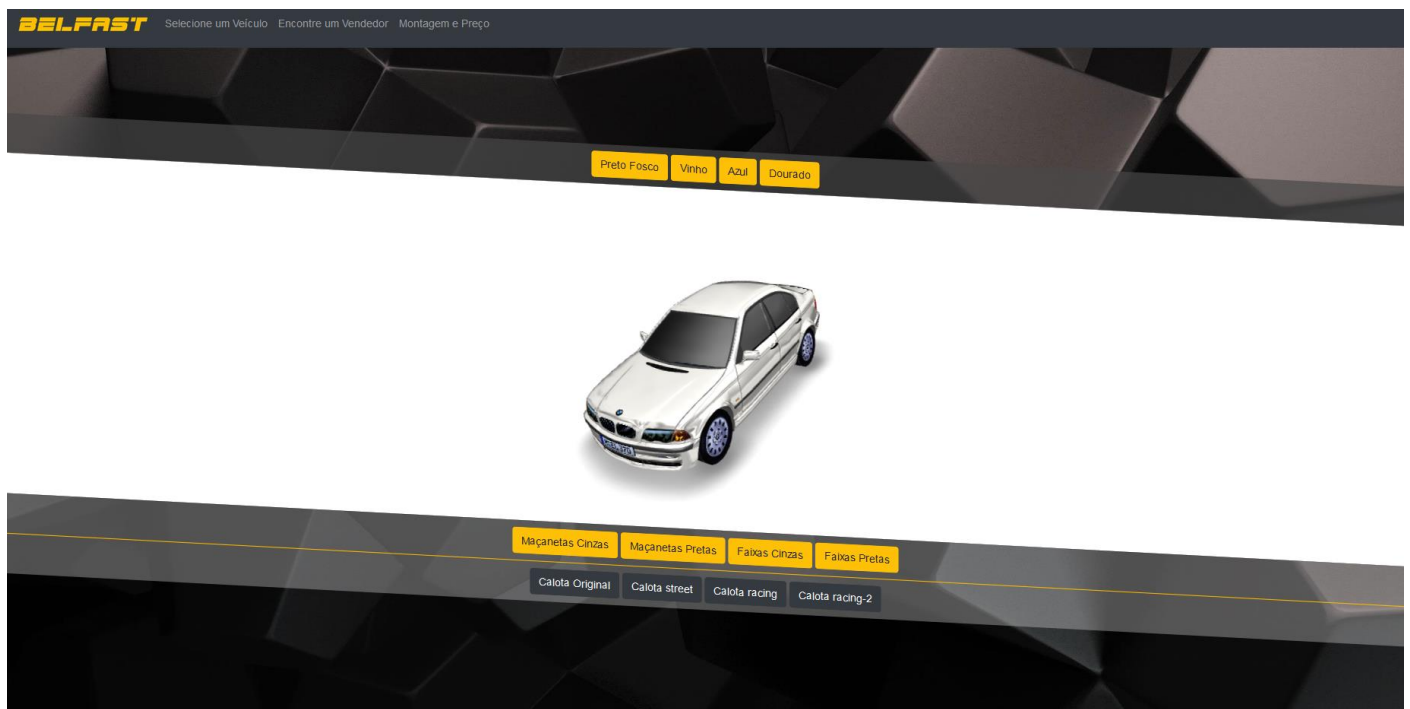
```
html, body {
  /*Footer colado no fim da página*/
  height: 100%;
}
```

Foi definido em ambos HTML e Body um height de 100% para assim deixando a altura da pagina sempre no maximo de resolução e mantendo os conteudo colocados a baixo, sempre abaixo, no caso o footer foi colado ao “chão” do website.

```
/*Resto da página*/
#main {
  min-height: 100%;
}
```

A altura mínima é sempre 100% de todo o conteúdo, esta propriedade foi atribuída a ID 'main' para que o corpo do website não deixe outra opção ao footer a não ser de ser o último conteúdo e sempre colado a baixo dos demais.

Resultado Final.



Desenvolvido por Diogo de Sousa Nogueira e Enrico Brosque Palma.

No protótipo foi utilizado algumas cores que não foram incluídas no projeto final, por falta de coerência com design do site.

As paletas de cores foram escolhidas por serem cores que não iriam agredir o design do site, assim mantendo o padrão de cores neutras, destaque no automóvel, responsivo e limpo.