



Relatório - ITEGRAFLIX - Processo Seletivo

Gabriel Augusto Sergio de Paula

gabrielaugustosergio@gmail.com

(31) 9 8808-5995

Neste relatório será listado, de forma detalhada e lúdica, os passos realizados para a codificação do projeto proposto.

As tecnologias utilizadas no projeto foram:

- **HTML**
- **CSS**
- framework **Bootstrap**, que fornece estilos CSS predefinidos.

HTML é um acrônimo para Hypertext (hipertexto) Markup (marcação) Language (linguagem), ela não é uma linguagem de programação em si, porém é uma linguagem na forma de escrever, tendo sintaxe e semântica, Hypertext é uma marcação específica que nos leva a outro texto, isso mais no passado, agora sendo relacionado a imagens, a vídeos e mais.

Já o CSS é uma linguagem de estilo, ou seja, define o layout do HTML.

Funcionalidades:

1. Primeiramente foi verificado os 10 principais filmes, segundo o The Movie DB. Após isso foi realizado o download das imagens necessárias;
 - a. A pasta img contém as imagens necessárias e os ícones.
2. A página principal é definida pelo arquivo “index.html”, neste arquivo html contém a lista dos filmes, exibindo o cartaz, nome do filme, nota, ano de lançamento e quantidade de votos recebidos; nas próximas subseções há uma explicação da codificação do documento html, no final do tópico há um resumo sobre a codificação;
 - a. A tag `<html lang="pt-br"></html>` é a principal tag, conhecida como tag raiz, tem um atributo `lang="pt-br"` indicando o principal idioma a ser usado na página. É importante para os navegadores fazerem o processamento mais rapidamente;
 - b. A tag `<head></head>` é a parte não visível pelo navegador, onde se faz configurações, contém informações como o título, links para o css, e mais;
 - c. `<meta>` ela serve para definir metadados, como codificação de caracteres especiais e portabilidade para dispositivos mobiles;
 - i. atributo `charset="UTF-8"` codificação de caracteres especiais;
 - ii. `name="viewport" content="width=device-width, initial-scale=1.0"` são os atributos que fornecem portabilidade para dispositivos mobiles; `width=device-width` indica que a largura da viewport deve corresponder à largura do dispositivo, enquanto `initial-scale=1.0` define o nível de zoom inicial da página;
 - iii. o atributo `name` para especificar a meta, `content` para conteúdo, como por exemplo `<meta name="author" content="Gabriel Augusto"; ou <meta name="description" content="Integraflix catálogos dos principais filmes">` o que vai aparecer no navegador ao buscar pelo site.
 - d. `<title>Integraflix</title>` título da página; na aba do navegador;
 - e. `<body></body>` Essa tag contém todo o conteúdo visual da página, como texto, imagens, entre outros;
 - i. `<header></header>` O cabeçario do site, o local onde fica a logo. tag semântica;
 - ii. `<nav></nav>` onde fica estruturado o menu;
 - i. contém um atributo `class="container"` classe do css que tem no bootstrap que faz a delimitação dos conteúdos ao centro e numa largura limite.
 - iii. a tag `<div></div>` é usada para agrupar um conteúdo. contém um atributo `class="itflix-mascot-container"` o `class` é um atributo para classificar as tags;
 - i. dentro da tag há uma outra tag `` contendo um atributo `src` ;
 - ii. `` é uma tag que define uma imagem. o `src` define o caminho ou URL da imagem a ser exibida. no caso, aponta para uma imagem na pasta “img”;
 - iii. `<h1></h1>` é usada para título, no caso, o ITEGRAFLIX que ficara no menu. A tag `<hx></hx` indo de 1 à 6, é a tag para título, 1 sendo o maior e 6 sendo o menor, a hierarquia indo de título, subtítulo e etc.

- f. `<link>` Para colocar o ícone usa a tag com o atributo `rel="icon"`, que seria para representar relação, no caso a tag link vai conter um ícone, depois o href para mostrar onde está o ícone;
- `<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-...
crossorigin="anonymous">` importa o arquivo CSS do Bootstrap a partir de um URL (caminho externo, o `href` especifica a localização. Os atributos `integrity` e `crossorigin` fornecem segurança;
 - `<link rel="stylesheet" href="css/styles.css">` tem quase que a mesma funcionalidade, só que importa um arquivo CSS personalizado chamado "styles.css" que está localizado na pasta local "css".
- g. A tag `<main></main>` a parte do site aonde vai o conteúdo principal, no caso, teremos lista de vários filmes. é utilizada para um conteúdo único da página, portanto, deve-se utilizá-la apenas uma vez por página, e deve ser colocada direto do body, como uma boa prática;
- dentro da tag `<main>` há uma tag `<section></section>` com um id e uma classe;
- A tag `<section>` serve para colocar seções na página HTML;
O atributo `id` é um identificar, logo é único, ou seja, é usado apenas 1 por Tag para identificação;
- h. Em seguida, é declarado um tag `<h2>` para definir um subtítulo, no caso, o texto "Populares";
- i. Uma tag `<div class="row">` No caso, a classe "row" é uma classe do Bootstrap que cria uma linha para conter colunas;
- j. Tem-se uma tag `<div class="col-sm-6 col-lg-4 col-xl-3 mb-3">`, dentro da tag `<div class="row">`, no qual "col-sm-6" indica que a coluna ocupará 6 colunas em dispositivos pequenos, "col-lg-4" ocupará 4 colunas em dispositivos grandes, "col-xl-3" indica que a coluna ocupará 3 colunas em dispositivos extra grandes, "mb-3" define uma margem inferior de 3 unidades. Tag a qual se repete para as demais tags div;
- k. Em seguida, é declarada outra `<div class="itflix-card">` dentro da div anterior, essa div contém a tag `<a>`, dentro da tag tem uma tag ``;
- a tag `<a>` conhecida como tag âncora, são os hyperlinks, recebe o atributo `href="details.html"`, a referência. No caso está redirecionando para a página "details.html", ao clicar na imagem;
 - a tag `` recebe atributo um `alt="Super Mario Bross"` que é um alternativo, ou seja, caso acontecer erro na exibição da imagem o texto alt será exibido, ficando "Super Mario Bross" no lugar da exibição.
- l. Após isso, é declarado a tag `<div class="itflix-card-description">`, a qual contém uma tag `<p>` e uma tag `<div class="itflix-stars">`, que contém uma imagem vetorizada e a tag `<h5>`, a div simboliza a avaliação. Tem-se outra tag `<h5>`. Após a tag tem-se outra tag `<div class="itflix-heart">`, que contém uma imagem vetorizada e a tag `<h5>`, a div simboliza a quantidade de curtidas.

Em suma, a estrutura da página HTML é organizada da seguinte forma:

- O arquivo "index.html" é o ponto de entrada da página principal;

- A lista de filmes é exibida dentro de uma seção principal, marcada pelo elemento `<main>`;
- Cada filme é representado por um elemento `<div>` contendo informações relevantes;
- O cartaz do filme é exibido através do elemento ``, utilizando o atributo `src` para especificar o caminho da imagem;
- O nome do filme é exibido utilizando o elemento `<p>`;
- A nota do filme é exibida utilizando o elemento `<div class="itflix-stars">`, contendo uma imagem de estrela e um elemento `<h5>` para exibir a nota;
- O ano de lançamento do filme é exibido utilizando o elemento `<h5>`;
- A quantidade de votos recebidos pelo filme é exibida utilizando o elemento `<div class="itflix-heart">`, contendo uma imagem de coração e um elemento `<h5>` para exibir a quantidade de votos;
- Essa é estrutura, a qual é repetida para exibir a lista completa, com o respectivo filme.

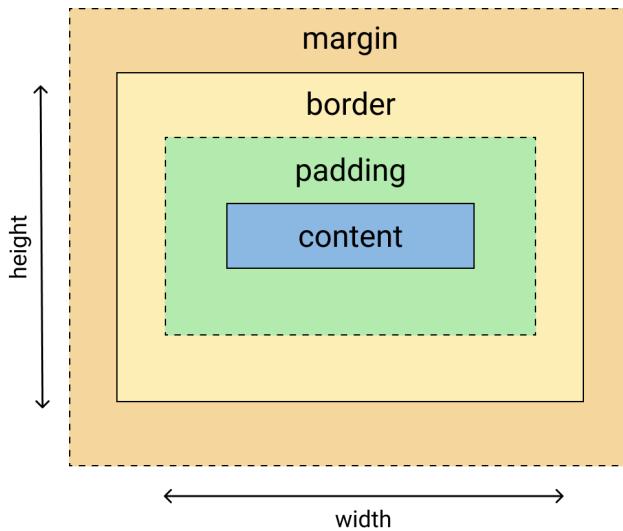
Dessa forma, o arquivo "index.html" estabelece a estrutura e o conteúdo da página principal, apresentando os filmes de forma organizada e informativa.

3. No arquivo style.css dentro da pasta CSS, responsável pela estilização da página; Os elementos são definidos por um nome de algum elemento HTML, são os seletores, uma chave aberta e uma fechada embaixo, e no meio tem-se as propriedades e seus valores. Toda propriedade é seguida de : (dois pontos) e um valor e um ; (ponto e vírgula) para encerrar essa ideia de valor. A seguir um exemplo:

```
h1, h2,
h3, h4,
h5, h6 {
    margin: 0;
}
```

- a. Primeiramente é importada, do serviço do Google Fonts, a fonte Roboto, que será a fonte utilizada nas páginas;
- b. `:root {}` seleciona o elemento raiz do documento, o que é definida as variáveis, isso é feito para ajudar na reutilização das cores. Na manutenção, por exemplo, se a cor que foi definida precisar ser modificada, ao invés de modifica-la uma a uma em todo o documento, basta modifica-la na raiz;
- c. O seletor universal `*` seleciona todos os elementos da página. Aqui, está sendo definido o modelo de caixa `border-box`, que inclui as bordas e o preenchimento no tamanho total do elemento. Além disso, a fonte Roboto está sendo definida como a fonte padrão para todos os elementos, caso a fonte Roboto não esteja disponível por algum motivo, uma alternativa 'sans-serif' é usada;
- d. Em seguida, é definido `hx{margin:0}`, regras se aplicam a todos os cabeçalhos (`h1` a `h6`) e estão removendo as margens padrão desses elementos;
 - i. o atributo `margin:0` é o espaço entre os elementos, espaços fora da caixa. Geralmente, é dividida em 4 valores `margin-top | margin-right | margin-bottom | margin-left` nesse caso está

sendo usada um shorthand, ou seja, todos os 4 valores são 0, de forma abreviada.



- e. A propriedade `background-color: var(--bg-gray);` que é definido no `html, body{}` define a cor de fundo do elemento selecionado;
 - i. `var(--bg-gray)` é uma das cores definidas no root.
- f. Após isso é definido um elemento `header{}`
 - i. o `color: var(--color-elements);` define a cor do texto no cabeçalho;
 - ii. o `height: 90px;` define a altura;
 - i. o `px` define a distância absoluta, ou seja, fixa e não altera o valor. A equivalência é $1\text{px} = 1/96\text{th of 1in}$;
 - ii. em alguns momentos da codificação será utilizado o `%`, define a distância relativa do viewport, logo uma maior adaptação aos diferentes tipos de tela.
 - iii. o `display: flex;` Aplica o modelo de layout flexível ao cabeçalho;
 - i. Permite posicionar os elementos dentro da caixa.
 - iv. o `align-items: center;` define o alinhamento, no caso alinha verticalmente os elementos dentro do cabeçalho ao centro;
 - v. o `border-bottom: outset 3px var(--color-elements);` são as bordas da caixa, propriedade que está abaixo da margin, no caso `outset` define o estilo da borda, o `3px` define a espessura da borda e o `var(--color-elements)` define a cor;
- g. Em seguida, é definido um elemento `nav {}`, consiste na modificação do elemento da barra do menu. O único atributo que tem no elemento que difere dos já abordados é o `justify-content: space-between;`
 - i. o `justify-content` comprehende uma estrutura de alinhamento dos itens no eixo principal, ou seja, alinhamento dos elementos dentro do container;

- i. o valor `space-between` cada elemento é disposto em um canto, um item no começo e o outro no fim.
- h. O `nav h1{font-size: 28px; font-weight: 600;}` modifica o elemento h1 dentro do elemento nav. Os atributos `font-size: 28px;` e `font-weight: 600;` compreendem a definições da fonte;
 - i. `font-size` Tamanho da fonte;
 - ii. `font-weight` Peso da fonte.
- i. O `.itflix-mascot-container{}` é um seletor de classe que se aplica a um contêiner que envolve o mascote da aplicação;
 - i. algo que ainda não foi abordado é o `.` (ponto) conhecido como class selector, um ponto seguido do nome da classe, para especificar qual parte do HTML se está modificando no CSS;
- j. O `.itflix-mascot-container img{margin-right: 10px; width: 60px;}` está modificando a imagem;
 - i. o `width` comprehende a definição da largura.
- k. Em seguida, o `#itflix-card-list {padding: 0 10px;}` é um ID selector, que é # (cerquilha, cardinal) e o ID do elemento HTML;
 - i. o `padding` é o preenchimento interno, ou seja, preenchimento dentro da caixa.
- l. A class selector `.itflix-stars h5 {color: var(--color-elements) !important;}` tem uma regra `!important` ela quebra o fluxo natural da cascata, define que essa cor será usada independente de qualquer regra que tente sobreescrita. Mas essa regra não é vista como uma boa prática;

Em suma, a codificação CSS:

- `margin` Define as margens externas de um elemento, controlando o espaço ao redor dele;
- `padding` Define o espaçamento interno de um elemento, controlando o espaço entre seu conteúdo e suas bordas;
- `display` Define o tipo de exibição do elemento, como "flex" para exibição flexível e "inline" para exibição em linha;
- `align-items` Controla o alinhamento vertical dos elementos filhos em um contêiner;
- `width` Define a largura de um elemento;
- `border-radius` Define o raio das bordas de um elemento, criando bordas arredondadas.
- `background-color` Define a cor de fundo de um elemento.
- `font-size` Define o tamanho da fonte de texto.
- `color` Define a cor do texto.
- `justify-content` Controla o alinhamento horizontal dos elementos filhos em um contêiner.
- Não foi mencionado nos tópicos, mas tem-se o atributo `min-height` o qual define a altura mínima de um elemento.

Os atributos definem estilos para diferentes elementos da página. Esses estilos contribuem para a aparência e o layout da página, definindo cores, tamanhos, espaçamentos e outros aspectos visuais dos elementos.