

Trabalhando com Componentes do Materialize e jQuery

MOSTRAR A UTILIZAÇÃO DE ALGUNS COMPONENTES DO FRAMEWORK MATERIALIZE PARA A CONSTRUÇÃO DE PÁGINAS WEB, UTILIZANDO TAMBÉM A BIBLIOTECA JQUERY.

AUTOR(A): PROF. EDSON MELO DE SOUZA

1. Introdução

O HTML, juntamente com o CSS e a biblioteca jQuery nos dão condições de construir sites dinâmicos. Entretanto, dependendo do projeto, é necessário que muitos códigos CSS sejam escritos, além do JavaScript.

Neste tópico veremos como utilizar as classes do *framework* Materialize para construção de páginas esteticamente atraentes e estruturadas, além de incluir dinamismo utilizando a biblioteca jQuery. Esse *framework* segue a mesma linha do Bootstrap, fornecendo classes para a formatação do documento HTML.

O objetivo deste tópico não é abordar o estudo de todos componentes contidos no Materialize, pois necessitaríamos de um livro para isso, mas sim, apresentar a utilização dos principais componentes, ou seja, os que mais são utilizados no dia a dia na criação de uma página web.

É importante lembrar que o sistema de funcionamento deste *framework* é muito semelhante ao Bootstrap e, portanto, de fácil assimilação, dispensando a abordagem sobre alguns conceitos.

2. Componentes Materialize

O Materialize fornece diversos componentes reutilizáveis para a criação de páginas como: ícones, elementos *dropdown* (suspensão), grupos de *input* (campos), barras de navegação, *cards* (caixas de mensagens) e muito mais. A seguir é apresentada uma lista dos componentes deste *framework*:

- Badges (Notificações)
- Buttons (Botões)
- Breadcrumbs (Rasteramento)
- Cards (Cartões)

- Chips (Blocos de informação)
- Collections (Agrupamento de objetos)
- Footer (Rodapé)
- Forms (Formulários)
- Icons (Ícones)
- Navbar (Barra de navegação)
- Pagination (Paginação)
- Preloader (Barra de progresso)
- Tabs (Abas)
- Tables (Tabelas)

Neste link (<http://materializecss.com> (<http://materializecss.com/>)) você pode acessar detalhadamente a criação e o funcionamento de cada componente acessando o link *Components*, na aba à esquerda do site. Os exemplos utilizados neste tópico foram retirados diretamente do site oficial do *framework* Materialize, entretanto, foram realizadas adaptações para melhorar a explicação.

Observação: Quando for necessário incluir um comentário, ele será feito diretamente no código, a fim de melhorar a compreensão.

Antes de iniciarmos a falar sobre os componentes, abaixo segue os códigos do arquivo HTML padrão que utilizaremos para desenvolver os exemplos. Portanto, em todos os exemplos serão mostrados apenas o código relativo ao que está sendo discutido, exceto quando orientado de forma diferente, pois há casos em que deveremos incluir códigos antes da criação do *container*.

Ao criar o arquivo do exemplo em seu computador, copie o código abaixo e então introduza o código apresentado na região onde está escrito “*seu código aqui*”.

```
1. <!DOCTYPE html>
2. <html>
3.     <head>
4.         <title>Componentes Materialize</title>
5.         <meta charset="UTF-8">
6.         <meta name="viewport" content="width=device-width, initial-scale=1">
7.
8.         <!-- Importação do Google Fonts -->
9.         <link href="http://fonts.googleapis.com/icon?family=Material+Icons">
10.
11.        <!-- Importação do CSS do framework -->
12.        <link type="text/css" rel="stylesheet" href="materialize/css/materialize.min.css">
13.
14.        <!-- Otimização para mobile -->
15.        <meta name="viewport" content="width=device-width, initial-scale=1">
16.
17.        <!-- Importação da biblioteca jQuery -->
18.        <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js">
19.
20.        <!-- Importação da biblioteca do Materialize -->
21.        <script type="text/javascript" src="materialize/js/materialize.min.js">
22.    </head>
23.    <body>
24.        <div id="container">
25.            seu código aqui
26.        </div>
27.    </body>
28. </html>
```

2.1 Menu de Navegação (nav-wrapper)

A usabilidade é um dos fatores que impactam diretamente na navegação de um usuário na internet e, por conta disso, um dos recursos mais utilizados para fornecer o conjunto de opções de acesso dentro de uma página são os “Menus ou Barras de Navegação”.

Existem diversos tipos de menus, podendo estar presentes no topo de uma página, nas laterais ou no rodapé e, em algumas situações, podem existir mais de um menu e até mesmo a repetição no rodapé da página, facilitando o acesso rápido ao usuário.

O Materialize possui uma classe que tem a função de criar menus de navegação muito bem estruturados e, para construir esses menus, utilizamos a classe “.nav-wrapper”. Os *nav-wrappers* são componentes meta responsivos que servem como cabeçalhos de navegação para seu aplicativo ou site. Eles começam colapsados ou recolhidos (e são alternáveis) em telas móveis e tornam-se horizontais conforme a largura da janela de exibição disponível aumenta.

As barras de navegação podem ser de diversos tipos e também podem conter elementos como links, botões, campos para digitação, imagens, entre outros. E, para cada elemento, existe uma classe que é utilizada em conjunto com a classe “.nav-wrapper”.

A seguir é mostrado o código para a criação de um menu simples fixo (estático).

```
1. <div id="container">
2.     <nav>
3.         <div class="nav-wrapper container">
4.             <a href="#" class="brand-logo">Logotipo</a>
5.             <ul id="nav-mobile" class="right hide-on-med-and-down">
6.                 <li><a href="#">Link 1</a></li>
7.                 <li><a href="#">Link 2</a></li>
8.                 <li><a href="#">Link 3</a></li>
9.             </ul>
10.        </div>
11.    </nav>
12. </div>
13. <div class="container">
14.    <p>Aqui vão os demais conteúdos textuais e elementos html</p>
15. </div>
```

A imagem a seguir mostra o código renderizado no navegador:



A imagem mostra um menu na parte superior da tela, com o nome da página (Logotipo) em destaque (negrito) na cor branca e três links denominados Link1, Link2 e Link3.

Você poderá encontrar mais tipos de navegação acessando o endereço <http://materializecss.com/navbar.html> (<http://materializecss.com/navbar.html>).

2.2 Cards (Painéis)

Os *cards* são componentes que tem a função de agrupar um conjunto de informações e são muito úteis para dar destaque. Por terem a característica de agrupar, podem conter outros componentes e elementos HTML em seu interior.

A utilização deste componente está relacionada a tag “card”, que possui diversas configurações para personalização de cores de fundo e do texto.

Vamos ver a seguir como criar um *card* utilizando a classe “.card”:

```

1. <div class="row">
2.     <div class="col s12 m5">
3.         <div class="card blue-grey darken-1">
4.             <div class="card-content white-text">
5.                 <span class="card-title">Título do Card (Painel)</span>
6.                 <p>Esse <em>card</em> foi criado com o fundo escuro e co
7.                 <p>Na parte inferior há dois links em destaque na cor la
8.             </div>
9.             <div class="card-action">
10.                <a href="#">Efetuar Login</a>
11.                <a href="#">Assinar</a>
12.            </div>
13.        </div>
14.    </div>
15.    <div class="col s12 m5">
16.        <div class="card">
17.            <div class="card-content">
18.                <span class="card-title">Título do Card (Painel)</span>
19.                <p>Esse <em>card</em> foi criado com o fundo branco e co
20.                <p>Na parte inferior há um painel na cor preta e com doi
21.            </div>
22.            <div class="card-action black">
23.                <a href="#">Efetuar Login</a>
24.                <a href="#">Assinar</a>
25.            </div>
26.        </div>
27.    </div>
28. </div>

```

A imagem a seguir mostra o resultado da criação de dois *cards* com conteúdo iguais, mas com configurações diferentes.



Você pode perceber que há várias possibilidades de construir um *card* e personalizar de acordo com a necessidade. Existem outros tipos de *cards* que aceitam imagens, vídeos, etc. Explore o componente estudando a documentação no endereço <http://materializecss.com/cards.html> (<http://materializecss.com/cards.html>).

2.3 Tabs (Abas)

Abas são um recurso muito utilizado em páginas na internet, pois sua construção facilita a tabulação ou agrupamento de dados de um mesmo tipo. Um exemplo disso é a divisão em grupos, por exemplo, de categoria de produtos.

A criação é simples e utiliza a classe “*tab*”, juntamente com a jQuery, que realiza modificações de diversas formas, proporcionando uma experiência muito interessante. A seguir vamos ver como criar um conjunto de abas que apresenta uma série de produtos.

O código a seguir contém, além da estrutura de criação das *tabs*, o *script* jQuery necessário para ativação do recurso.

```
1. <!DOCTYPE html>
2. <html>
3.     <head>
4.         <title>Componentes Materialize</title>
5.         <meta charset="UTF-8">
6.         <meta name="viewport" content="width=device-width, initial-scale=1">
7.
8.         <!-- Importação do Google Fonts -->
9.         <link href="http://fonts.googleapis.com/icon?family=Material+Icons">
10.
11.        <!-- Importação do CSS do framework -->
12.        <link type="text/css" rel="stylesheet" href="materialize/css/materialize.min.css">
13.
14.        <!-- Otimização para mobile -->
15.        <meta name="viewport" content="width=device-width, initial-scale=1">
16.
17.        <!-- Importação da biblioteca jQuery -->
18.        <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js">
19.
20.        <!-- Importação da biblioteca do Materialize -->
21.        <script type="text/javascript" src="materialize/js/materialize.min.js">
22.
23.        <script type="text/javascript">
24.            $(document).ready(function () {
25.                $('ul.tabs').tabs('select_tab', 'tab_id');
26.            });
27.        </script>
28.    </head>
29.
30.    <body>
31.        <div class="container">
32.            <h1>Nossos Produtos</h1>
33.            <div class="row">
34.                <div class="col s12">
35.                    <ul class="tabs">
36.                        <li class="tab col s3"><a class="active" href="#sala">Sala</a>
37.                        <li class="tab col s3"><a href="#cozinha">Cozinha</a>
38.                        <li class="tab col s3 disabled"><a href="#games">Games</a>
39.                        <li class="tab col s3"><a href="#musica">Música</a>
```



```

40.         </ul>
41.     </div>
42.     <div id="eletronicos" class="col s12">Você está em Eletr
43.     <div id="cozinha" class="col s12">Você está em Cozinha</
44.     <div id="games" class="col s12">Item desabilitado</div>
45.     <div id="musica" class="col s12">Você está em música</di
46.     </div>
47. </div>
48. </body>
49. </html>

```

Nas linhas de 23 a 26 a jQuery é utilizada para quando houver uma seleção no componente, realizar sua expansão.

Na imagem a seguir são mostradas as abas criadas com base no código anterior. Você pode visualizar as abas e, quando uma aba está selecionada, ela recebe uma linha (*underline*) em seu nome, mostrando que há uma seleção naquele elemento.



2.4 Form (Formulários)

Não ter um formulário em uma página do site significa que você não está dando a atenção necessária ao seu visitante, pois, esta é a forma mais prática de manter contato com o usuário, de forma que ele possa preenche-lo e lhe enviar.

Outro ponto é a criação do cadastro, onde é necessário o preenchimento de campos. Outra situação é o meio de acesso a áreas reservadas, os quais fazemos utilizando uma tela de login.

O Materialize possui um conjunto de classes para serem utilizadas em formulários, o que faz com que sua aparência fique muito atraente e também funcional. Diferente do Bootstrap, esse *framework* trabalha diretamente nos elementos HTML e não no formulário como um todo.

Dentro deste item vamos tratar também dos campos de entrada ou os “*inputs*”, pois podemos aplicar classes que farão uma transformação substancial na utilização, eliminando, por exemplo, o uso dos “*labels*”, como também a aplicação de ícones nos campos, o que facilita muito a visualização para o usuário.

No código a seguir é criado um formulário simples de cadastro que utilizará classes diversas para personaliza-lo. Vamos então ao código:

```
1. <!DOCTYPE html>
2. <html>
3.     <head>
4.         <title>Componentes Materialize</title>
5.         <meta charset="UTF-8">
6.         <meta name="viewport" content="width=device-width, initial-scale=1">
7.
8.         <!-- Importação do Google Fonts -->
9.         <link href="http://fonts.googleapis.com/icon?family=Material+Icons">
10.
11.         <!-- Importação do CSS do framework -->
12.         <link type="text/css" rel="stylesheet" href="materialize/css/materialize.min.css">
13.
14.         <!-- Otimização para mobile -->
15.         <meta name="viewport" content="width=device-width, initial-scale=1">
16.
17.         <!-- Importação da biblioteca jQuery -->
18.         <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js">
19.
20.         <!-- Importação da biblioteca do Materialize -->
21.         <script type="text/javascript" src="materialize/js/materialize.min.js">
22.     </head>
23.
24.     <body>
25.         <div class="container">
26.             <h4>Formulário de Cadastro</h4><br>
27.             <div class="row">
28.                 <form class="col s4">
29.                     <div class="row">
30.                         <div class="input-field col s12">
31.                             <i class="material-icons prefix">account_circle</i>
32.                             <input placeholder="informe seu nome" id="nome">
33.                             <label for="nome">Nome</label>
34.                         </div>
35.                     </div>
36.                     <div class="row">
37.                         <div class="input-field col s12">
38.                             <i class="material-icons prefix">phone</i>
39.                             <input placeholder="informe seu telefone" id="telefone">
```

```

40.         <label for="telefone">Telefone</label>
41.     </div>
42. </div>
43. <div class="row">
44.     <div class="input-field col s12">
45.         <i class="material-icons prefix">vpn_key</i>
46.         <input placeholder="informe sua senha" id="p
47.         <label for="password">Senha</label>
48.     </div>
49. </div>
50. <div class="row">
51.     <div class="input-field col s12">
52.         <i class="material-icons prefix">email</i>
53.         <input id="email" type="email" class="valida
54.         <label for="email">Email</label>
55.     </div>
56. </div>
57. <div class="row">
58.     <div class="input-field col s12">
59.         <input type="checkbox" id="test5" />
60.         <label for="test5">Receber informativos</lab
61.     </div>
62. </div>
63. <div class="row">
64.     <div class="input-field col s12">
65.         <button type="submit" class="btn">Enviar</bu
66.         <button type="submit" class="btn blue-grey">
67.     </div>
68. </div>
69. </form>
70. </div>
71. </div>
72. </body>
73. </html>

```

A seguir é mostrada a imagem resultante da renderização do código acima, onde pode ser visualizado o formulário criado com marcações visuais ao lado de cada campo. Esse recurso é muito interessante, pois elimina a utilização de “*labels*”.



← → ↻ ⓘ localhost/ComponentesMaterialize/

Formulário de Cadastro

 Nome
informe seu nome

 Telefone
informe seu telefone

 Senha
informe sua senha

 Email
informe seu e-mail

☐ Receber informativos

ENVIAR **CANCELAR**

Quando o código é executado no navegador, é possível visualizar a troca de cor quando um campo recebe o foco, o que dá uma dinâmica diferente apresentada ao usuário.

Você pode perceber o quão é poderoso o *framework* Materialize, portanto, vale muito investir seu tempo e aprender a utilizar outros recursos e componentes, além dos apresentados, para criar páginas profissionais.

A seguir veremos um exemplo onde a biblioteca jQuery é utilizada para dar dinamismo aos controles.

3 Aplicando Efeitos no Materialize com a Biblioteca jQuery

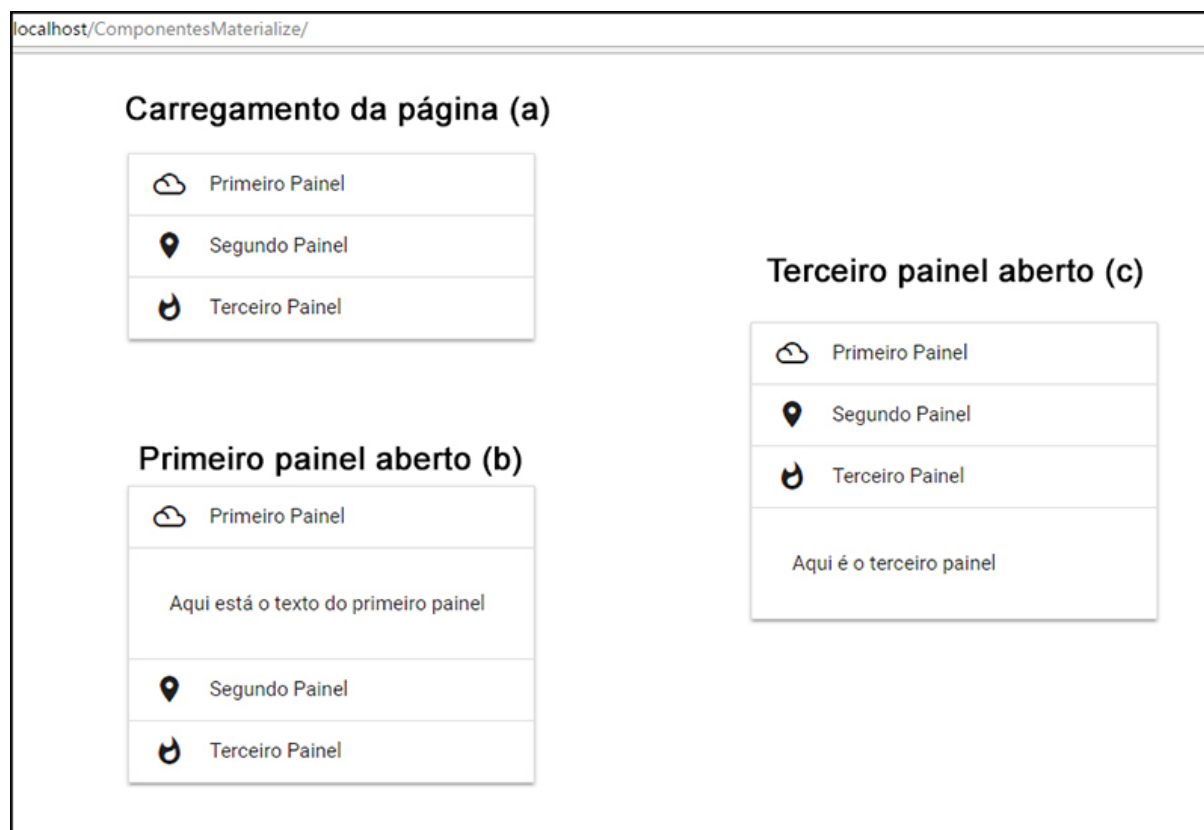
Apesar de já termos utilizado a biblioteca jQuery em exemplos anteriores, vamos criar um painel (*Collapsible*) que disponibilizará a funcionalidade de “abrir e fechar”, como em um efeito cortina. Este painel contém algum texto em seu interior e, quando clicado, expande e mostra o conteúdo em seu interior.

Ao carregar a página, o painel estará fechado e, ao clicar sobre o título, ele se abrirá. Será necessário a criação de um script para a jQuery disponibilizar os efeitos para interagirmos. Uma característica interessante deste *framework* é que ele já dispõe de classes que permitem incluir ícones sem a necessidade da criação de um CSS específico para isso.

Vamos então ao código:

```
1. <div class="row">
2.     <div class="col s4">
3.         <ul class="collapsible" data-collapsible="accordion">
4.             <li>
5.                 <div class="collapsible-header"><i class="material-icons"
6.                 <div class="collapsible-body"><span>Aqui está o texto do
7.             </li>
8.             <li>
9.                 <div class="collapsible-header"><i class="material-icons"
10.                <div class="collapsible-body"><span>Aqui é o segundo pai
11.            </li>
12.            <li>
13.                <div class="collapsible-header"><i class="material-icons"
14.                <div class="collapsible-body"><span>Aqui é o terceiro pa
15.            </li>
16.        </ul>
17.    </div>
18. </div>
```

A seguir é mostrada a imagem após a renderização do código.



A imagem anterior mostra as três situações, uma quando carrega a página que o painel está fechado (a), na sequência (b) é aberto o primeiro painel e, por fim (c) o terceiro painel é aberto e os demais fechados. Aqui finalizamos este tópico que mostrou alguns recursos dos componentes do Materialize. Mas, não se esqueça que há muito mais e, portanto, você deve acessar os links de referências a seguir para saber mais sobre as classes mostradas e sobre outras classes disponíveis.

SAIBA MAIS!

Materialize Oficial – <http://www.materializecss.com> (<http://www.materializecss.com/>)

Materialize Tutorial: <http://www.tutorialspoint.com/materialize/>
(<http://www.tutorialspoint.com/materialize/>)

Resumo

Neste tópico foram abordadas algumas das classes mais utilizadas na construção de páginas, descrevendo sua funcionalidade e implementando exemplos. Além das classes, também foi apresentado um exemplo que utiliza a jQuery para implementar dinamismo na página.

Conclusão

O *framework* Materialize abre quase que infinitas possibilidades na construção de páginas Web e também para dispositivos móveis. É importante que seja dedicado um tempo para conhecer melhor os detalhes sobre as classes disponíveis no Materialize, bem como explorar a utilização da jQuery para implementar funcionalidades dinâmicas na construção de páginas web.

Diferente do Bootstrap, o Materialize permite a inclusão de componentes com menos código, apesar do Bootstrap possuir mais componentes e personalizações de forma direta.

ATIVIDADE FINAL

Para criar uma barra de navegação simples é necessária a utilização da classe:

- A. nav-wrapper.
- B. navbar s12.
- C. nav-pagination.
- D. nav-content.

A classe ".card" tem a função de:

- A. Criar um cartão ou um grupo.
- B. Criar um elemento para navegação em painéis.
- C. Configurar a utilização do Materialize.
- D. Informar que será incluído um script para jQuery.

A classe que permite incluir um ícone para um campo "input" é:

- A. icon_prefix.
- B. linkrel.
- C. form-group.
- D. col-sd-12

REFERÊNCIA

LEARNING MATERIALIZE. Tutorials Point. Disponível em: <<http://www.tutorialspoint.com/materialize> (<http://www.tutorialspoint.com/materialize>)>. Acesso em 21 de mai. 2017.

MATERIALIZE CSS. Aprenda sobre o Material Design e o nossa Equipe do Projeto. Disponível em: <<http://materializecss.com/about.html> (<http://materializecss.com/about.html>)>. Acesso em: 24 mai. 2017.

MEYER, J. O Guia Essencial do HTML 5: usando jogos para aprender HTML 5 e JavaScript. Rio de Janeiro: Ciência Moderna, 2011.

RICHARDS, G.; LEBRESNE, S.; BURG, B.; VITEK, J. An analysis of the dynamic behavior of JavaScript programs. *2010 ACM SIGPLAN conference on Programming language design and implementation PLDI 10*, v. 45, n. 6, p. 1, 2010.

UNINOVE. Tecnologia para Navegadores. Material AVA - Aplicações Web. 2012.

VAUGHAN-NICHOLS, Steven J. Will HTML 5 restandardize the web? *Computer*, v. 43, n. 4, p. 13-15, 2010.

W3C. HTML, The Webs Core Language. Disponível em: <<http://www.w3.org/html/>