

Adicionando Menu de Navegação

Transcrição

Uma vez que nossas páginas estão melhores, podemos finalizar as alterações de layout da administração colocando um Menu de navegação.

Dentro do bootstrap, vá para a página <http://getbootstrap.com/examples/starter-template/> (<http://getbootstrap.com/examples/starter-template/>) e vamos olhar novamente o código fonte, pegando a parte de navegação (identificada pelas tags `<nav>`). Dessa parte de navegação, vamos manter apenas o que é importante para nosso sistema.

Vá para o formulário (`form.xhtml`), e logo antes da div `<div class="container">`, adicione o código abaixo:

```
<nav class="navbar navbar-inverse navbar-fixed-top">
  <div class="container">
    <div class="navbar-header">
      <a class="navbar-brand" href="#">Casa do Código</a>
    </div>
    <div id="navbar" class="collapse navbar-collapse">
      <ul class="nav navbar-nav">
        <li><h:link value="Lista" outcome="lista" /></li>
        <li class="active"><a href="#">Cadastro</a></li>
      </ul>
    </div><!-- .nav-collapse -->
  </div>
</nav>
```

Você deve ter percebido que no link para a lista, usamos uma tag diferente: `<h:link>` . No JSF podemos fazer links de forma mais simples, apenas dizendo o nome da página XHTML de onde queremos ir, `outcome` , sem informar a extensão. Ficando apenas:

```
<h:link value="Lista" outcome="lista" />
```

O `value` é o nome que será exibido no link.

Rode a aplicação e veja o resultado na tela. Perceba que perdemos um pouco do conteúdo da página, pois o atual menu está cobrindo o conteúdo. Existem várias formas de resolver isso, mas uma forma elegante e que não impacte a nossa aplicação de modo geral é criando um novo arquivo CSS para controlar nossa administração.

Crie um novo arquivo CSS chamado `style-admin.css` e coloque-o dentro da pasta `resources/css` .

Nesse arquivo vamos definir alguns estilos, começando pela tag `body` . Nela precisamos definir um `padding-top` com o valor de `80px` . Além disso, vamos deixar um espaço no final da página também, deixando todo o conteúdo bem livre no final do navegador. Para isso, usaremos o `padding-bottom` e o valor dele de `20px` . Assim, nosso `style-admin.css` ficará assim:

```
body {
    padding-top: 80px;
    padding-bottom: 20px;
}
```

Voltando no `form.xhtml`, dentro da tag `<h:head>` vamos adicionar outra tag `<h:outputStylesheet>`, onde o `library` terá o valor `css` e o `name` dessa vez o valor do nosso novo arquivo `style-admin.css`.

Para o nosso cadastro de livros ficar ainda mais bacana, vamos adicionar um título a essa página, facilitando a localização do usuário dentro do nosso sistema. Vamos adicionar uma tag `<h1>` logo após o `<div class="container">` e logo antes da declaração do `<h:form>`. O código é bem simples:

```
<h1>Cadastro de Livros</h1>
```

Ainda precisamos fazer a mesma coisa para a nossa listagem. Abra o arquivo `lista.xhtml` e para não repetir toda explicação e como o código é exatamente o mesmo, vamos copiar de lá e já colocar aqui. Fique atento para as mudanças, pois parte do código já tínhamos:

```
<h:head>
    <h:outputStylesheet library="css" name="bootstrap.min.css" /> <!-- JÁ TINHA -->
    <h:outputStylesheet library="css" name="style-admin.css" /> <!-- ADICIONAMOS AGORA -->
</h:head>

<h:body>
    <nav class="navbar navbar-inverse navbar-fixed-top">
        <div class="container">
            <div class="navbar-header">
                <a class="navbar-brand" href="#">Casa do Código</a>
            </div>
            <div id="navbar" class="collapse navbar-collapse">
                <ul class="nav navbar-nav">
                    <li class="active"><a href="#">Lista</a></li>
                    <li><h:link value="Cadastro" outcome="form" /></li>
                </ul>
            </div><!--/.nav-collapse -->
        </div>
    </nav>

    <div class="container">
        <h1>Lista de Livros</h1>
        <h:messages />
    </div>
<!-- CÓDIGO CONTINUA COM A DATATABLE -->
</div>
</h:body>
```

Vamos adicionar mais um `class` na nossa tabela, abra o `<h:dataTable>` e no `class` coloque mais um `class` que será o `table-hover`. Ele ativa uma cor `"cinzada"` quando o usuário passa o mouse por cima da nossa tabela.

Com isso, deixamos nossas páginas bem mais bacana para utilizar na administração do nosso sistema. A experiência do usuário (**UX**) é sempre importante, e temos até mais prazer em usar um sistema bonito que se apresenta fácil de usar.

Fechamos as melhorias de UX da administração do sistema, e com isso vamos para os exercícios. Qualquer dúvida, você já sabe, utilize o fórum. Bons estudos.