

03

Internacionalização

Transcrição

Mas da forma que fizemos, acabamos perdendo a quantidade de itens no carrinho que era exibido no menu. A solução é passar esse valor por parâmetro. No `messages.properties` teremos:

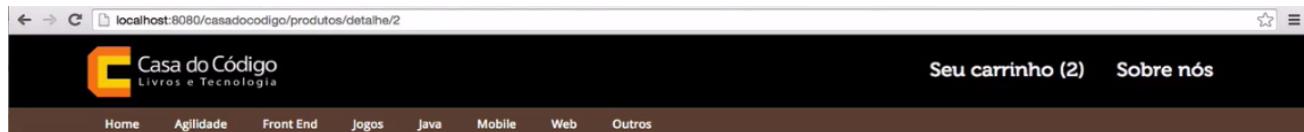
```
[...]
menu.carrinho = Seu carrinho ({0})
[...]
```

Observação: O termo `{0}` indica que estamos recebendo um parâmetro para ser exibido com a mensagem e o número indica que queremos o parâmetro naquela posição, `0` no caso, seria o primeiro parâmetro.

E para passar este parâmetro, precisamos mudar a tag `fmt:message` para que em seu corpo possamos usar a tag `fmt:param` especificando o valor do parâmetro.

```
<li>
    <a href="${s:mvcUrl('CCC#itens').build()}" rel="nofollow">
        <fmt:message key="menu.carrinho">
            <fmt:param value="${carrinhoCompras.quantidade}" />
        </fmt:message>
    </a>
</li>
```

E assim teremos o contador de itens no carrinho de volta:



O que fizemos até aqui foi simplesmente traduzir algumas mensagens. Vamos internacionalizar realmente. Vamos por mensagens em Inglês agora. Crie um novo arquivo de mensagens e nomeie este arquivo como `messages_en_US.properties` e cole as mensagens já traduzidas a seguir:

```
field.required = Required field
field.required.product.titulo = Field title is required
field.required.product.numeros = Field pages is required
field.required.product.descricao = Field description is required
typeMismatch = The data type is invalid
typeMismatch.product.numeros = Type a valid value: Example "100"
```

```
navegacao.categoria.home = Home
navegacao.categoria.agile = Agile
navegacao.categoria.front_end = Front End
navegacao.categoria.games = Games
navegacao.categoria.java = Java
navegacao.categoria.mobile = Mobile
navegacao.categoria.web = Web
navegacao.categoria.outros = Others
```

```
menu.carrinho = Shopping Cart ({0})
menu.sobre = About us
```

Observação: Note que o nome do arquivo tem um novo sufixo (`en_US`) que diz respeito ao **Locale** do usuário. Originalmente o *locale* é separado por traço (`en-US` ou `pt-BR`), mas geralmente as linguagens de programação traduzem o traço para o *underline* quando usados recursos de localização.

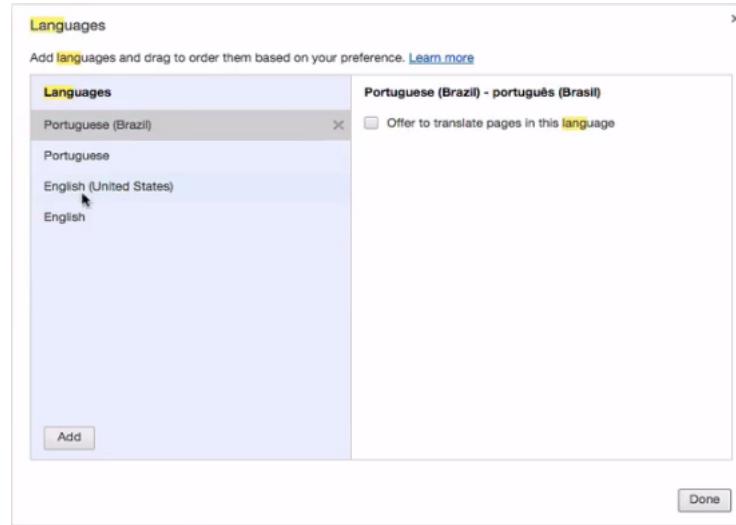
Agora nosso cabeçalho aparece totalmente em Inglês mesmo que nossas configurações do navegador estejam em Português.



Isso acontece por que o nosso navegador não solicita somente que a página venha em português, mas em inglês também. Se verificarmos nos cabeçalhos de requisição nas ferramentas do desenvolvedor, veremos que o navegador neste caso, aceita que as páginas sejam enviadas tanto em português quanto em inglês.

```
Request Headers
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Encoding: gzip, deflate, sdch
Accept-Language: pt-BR,pt;q=0.8,en-US;q=0.6,en;q=0.4
Cache-Control: max-age=0
Connection: keep-alive
```

E se verificarmos nas configurações, teremos os idiomas que o navegador suporta para configurar um ou outro como padrão.



Por nosso navegador aceitar diversos idiomas, o *Spring* irá responder com o idioma que encontrar primeiro. Ele deveria responder em português já que é o primeiro idioma na lista da requisição, mas não, por que? Por que o *Spring* está buscando por um arquivo com o sufixo `pt_BR`.

Como não encontra, o *Spring* passa para o próximo da lista, neste caso, somente `pt`, mas também não encontra e então encontra o `en_US` com o qual responde. O `messages.properties` sem sufixo só irá ser usado em último caso. O que podemos fazer então para que nosso cabeçalho volte a ficar em português? Simples, duplicaremos o arquivo `messages.properties` e adicionaremos o sufixo `pt`. Assim o arquivo se chamará `messages_pt.properties`.

Para exibição de mensagens parametrizadas, como a do carrinho de compras por exemplo, existe uma forma mais curta de exibi-la. Usando a tag `s:message` do *Spring* podemos por meio do atributo `code` definir qual é a chave da mensagem a ser exibida e através do atribuído `arguments` definir o valor do parâmetro esperado.

A exibição do texto do menu do carrinho de compras pode passar desta forma:

```
<fmt:message key="menu.carrinho">
    <fmt:param value="${carrinhoCompras.quantidade}" />
</fmt:message>
```

Para esta forma:

```
<s:message code="menu.carrinho" arguments="${carrinhoCompras.quantidade}" />
```