

## Internacionalização e validação

### Capítulo 4 - Internacionalização e validação

Já temos diversas features implementadas em nosso site. Mas e se nosso sistema crescer e quisermos que ele funcione em outros países? Não é uma boa prática criarmos diversos sites, cada um com uma língua. Se quisermos adicionar uma nova funcionalidade, teríamos que mexer no código de todos.

Existem traduções disponíveis para os textos que estamos utilizando e o Cake PHP possui um componente de **internacionalização**, que possibilita, de uma forma fácil, modificarmos o idioma da aplicação. Para isso precisaremos fazer alguns ajustes.

Vamos entrar no arquivo "src/Controller/ContatoController". Perceba que passamos mensagens em português, como "Email enviado com sucesso". O Cake PHP pode traduzir esses textos. Para isso precisamos defini-los chamando a função `__()`:

```
...
$this->Flash->set(__('Email enviado com sucesso'), ...
$this->Flash->set(__('Falha ao enviar o email'), ...
```

Continuemos fazendo o mesmo para as outras mensagens do site:

```
__("Produto removido com sucesso");
__("Erro ao deletar o produto");
__("Produto salvo com sucesso");
__("Erro ao salvar o produto");
```

Só definimos as mensagens de validação. Temos ainda muito texto na interface do site, como os campos dos produtos e os links. AOu seja, precisaremos aplicar a função nos textos das Views. Como elas estão em HTML, precisamos abrir a tag do PHP:

```
<th><?= __('Id'); ?></th>
<th><?= __('Nome'); ?></th>
<th><?= __('Preço'); ?></th>
<th><?= __('Preço com desconto'); ?></th>
<th><?= __('Descrição'); ?></th>
<th><?= __('Ações'); ?></th>
```

E assim por diante para todos os textos. Lembrando-se que onde não houver a tag do PHP, ela deve ser incorporada. Feito isso, todos os textos da nossa aplicação estão prontos para serem traduzidos. Agora precisamos passar para o Cake PHP que todos os textos que estão dentro da função `__()` devem ser traduzidos.

Vamos ao Terminal, dentro do diretório da nossa aplicação e fazemos:

```
bin/cake i18n extract
```

O `i18n` é uma abreviação para *internationalization* e pedimos para extrair os textos que queremos traduzir. Ao executar, o Cake perguntará

- se queremos extrair realmente
- qual o caminho de extração
- também é possível extrair mensagens internas, como erro 404. Neste caso, falaremos que não
- ainda escolhemos o caminho dos arquivos de tradução (já vem um indicado por padrão)
- se queremos dar *merge* nas strings dentro de um arquivo default.po

Podemos ver que dentro do diretório src foi criada a pasta Locale e, dentro dela, o arquivo default.po.

Vamos agora testar traduzindo nossa aplicação para o inglês. Dentro da pasta Locale criamos outra com o nome "en\_US". Copiamos o arquivo "default.po" para dentro dessa nova pasta. Como regra, mudamos a extensão do arquivo para .po: "default.po" e será este arquivo a ser traduzido. Toda as mensagens estarão lá. Basta adicionar suas traduções para o inglês. Use dicionários ou ferramentas de tradução da internet para tal. Veja um exemplo:

```
#: Controller/ContatoController.php:12
msgid: "Email enviado com sucesso"
msgstr "Email successfully sent"
```

Atualizando a página:

O site foi traduzido para o inglês. Mas como o navegador sabe que queremos utilizar o idioma em inglês? Se abrirmos o arquivo "config/bootstrap.php", podemos perceber que o idioma padrão está setado para o inglês:

```
ini_set('intl/default_locale', 'en_US');
```

Podemos modificar esta linha para que o padrão fique em português do Brasil:

```
ini_set('intl/default_locale', 'pt_BR');
```

Assim, ao atualizarmos novamente a página, ela volta para o português. E para traduzir para outros idiomas, seguimos o mesmo processo.

