



Desenvolvimento Web II

Framework Laravel 5

*Rotas / Parâmetros / Eloquent
(Curso e Turma)*

Gil Eduardo de Andrade





Framework Laravel

Parâmetros e Rotas

- Ao definirmos as rotas da aplicação dentro Laravel (arquivo web.php) é possível especificarmos que, ao final do caminho, um valor será passado como parâmetro. Exemplo:
 - ***Route::get('/cursos/editar/{id}', 'CursoController@editar');***

(Observe a sintaxe “*{id}*”, ela indica que um parâmetro nomeado como “*id*” será passado via url e recebido pelo **método editar** da classe **CursoController.php**)





Framework Laravel

Parâmetros e Rotas

- Considerando o exemplo anterior, onde o método *editar* é invocado para dar início a alteração dos dados de um curso previamente selecionado pelo usuário;
- Torna-se necessário então (ao método), receber o identificador deste curso, que lhe permitirá obter do banco as informações do mesmo;



Framework Laravel

Exemplo Editar: (código)

Arquivo "web.php"

```
Route::get('/cursos', 'CursoController@listar');
Route::get('/cursos/cadastrar', 'CursoController@cadastrar');
Route::get('/cursos/editar/{id}', 'CursoController@editar');
Route::post('/cursos/salvar/{id}', 'CursoController@salvar');
```

Define as rotas para editar (*/editar/{id}*) e salvar (*/salvar/{id}*) as informações do curso, indicando que elas recebem um parâmetro *{id}*.

Arquivo "CursoController.php" – método "editar(\$id)"

```
public function editar($id) {

    $curso = CursoModel::find($id);
    // Verifica se existe um curso com o 'id' recebido por parâmetro
    if(empty($curso)) {
        return "<h2>[ERRO]: Curso não encontrado para o ID=" . $id . "!</h2>";
    }
    return view('cursoEditar')->with('curso', $curso);
}
```

O método editar recebe o parâmetro *{id}* e utiliza-o para buscar (no banco) as informações relativas ao curso em questão. Observe que o método *find()* do *Eloquent* foi utilizado, ele funciona como uma cláusula *WHERE* do *SQL*, filtrando as informações via *id*.

As informações obtidas são armazenadas na variável *\$curso* e enviadas para *view cursoEditar.blade.php*.

Framework Laravel

Exemplo Editar: (código)

Arquivo "cursoEditar.php"

```
<form action="{ action('CursoController@salvar', $curso->id) }" method="POST">
  <input type="hidden" name="_token" value="{{ csrf_token() }}">

  <label>Nome: </label>
  <input type="text" name="nome" value="{{ $curso->nome }}" class="form-control">
  <br>
  <label>Abreviatura: </label>
  <input type="text" name="abreviatura" value="{{ $curso->abreviatura }}" class="form-control">
  <br>
  <button type="submit" class="btn btn-success btn-block">Salvar</button>
</form>
```



Utilizando os recursos do *blade* acessamos os dados armazenados em *\$cursos* para montar o formulário de edição que contém as informações editadas e posteriormente salvas pelo usuário. Observe que o *action do form* aponta para o método salvar e passa como parâmetro o *id do curso* que terá suas informações alteradas.



Framework Laravel

Exemplo Editar: (código)

Arquivo “CursoController.php” – método “salvar(\$id)”

```
// INSERT
if($id == 0) {
    $objCursoModel = new CursoModel();
    $objCursoModel->nome = mb_strtoupper(Request::input('nome'), 'UTF-8');
    $objCursoModel->abreviatura = mb_strtoupper(Request::input('abreviatura'), 'UTF-8');
    $objCursoModel->save();
}
// UPDATE
else {
    $objCursoModel = CursoModel::find($id);
    $objCursoModel->nome = mb_strtoupper(Request::input('nome'), 'UTF-8');
    $objCursoModel->abreviatura = mb_strtoupper(Request::input('abreviatura'), 'UTF-8');
    $objCursoModel->save();
}
```



Na implementação utilizada o método salvar é utilizado para cadastro e alteração de cursos. Sendo assim, repare que para alteração (*else*) usamos o método *find()* para encontrar o curso com *id=\$id* no banco, alteramos as informações (*nome e abreviatura*) desse curso (*objeto*) e invocamos o método *save()* para confirmar essas alterações na base de dados.

Framework Laravel

Exemplo Editar: (execução)

Arquivo "curso.blade.php"



[Home](#)

BD / Migration / Seeder / Model / Controller



Cursos Cadastrados

[Cadastrar Novo Curso](#)

ID	NOME DO CURSO	ABREVIATURA	EVENTOS
1	ANÁLISE E DESENVOLVIMENTO DE SISTEMAS	TDAS	Editar Remover
2	EMI EM INFORMÁTICA	INFO	Editar Remover
4	PÓS-GRADUAÇÃO EM MATEMÁTICA COMPUTACIONAL	PGMC	Editar Remover

Selecione a opção **Editar**.

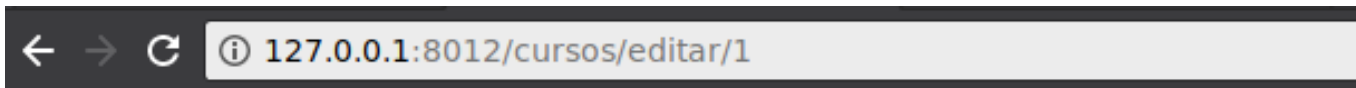
©2018 » Gil Eduardo de Andrade



Framework Laravel

Exemplo Editar: (execução)

Arquivo “*cursoEditar.blade.php*”



[Home](#)

BD / Migration / Seeder / Model / Controller



Alterar Dados do Curso

Nome:

Abreviatura:

©2018 » Gil Eduardo de Andrade

Informações carregadas para o curso selecionado.

Botão para salvar alterações, invoca o método *salvar(\$id)* da classe *CursoController.php*.



Framework Laravel

Exemplo Remover: (código)

Arquivo "web.php"

```
Route::get('/cursos', 'CursoController@listar');
Route::get('/cursos/cadastrar', 'CursoController@cadastrar');
Route::get('/cursos/editar/{id}', 'CursoController@editar');
Route::post('/cursos/salvar/{id}', 'CursoController@salvar');
Route::post('/cursos/remover/{id}', 'CursoController@remover');
```

Define a rotas para remover (*/remover/{id}*) o curso, indicando que ela recebe um parâmetro *{id}*.

Arquivo "CursoController.php" – método "remove(\$id)"

```
public function remove($id) {

    $objCursoModel = CursoModel::find($id);
    // Verifica se existe um curso com o 'id' recebido por parâmetro
    if(empty($objCursoModel)) {
        return "<h2>[ERRO]: Curso não encontrado para o ID=".$id."</h2>";
    }
    $objCursoModel->delete();

    return redirect()->action('CursoController@listar')->withInput();
}
```

O método remove recebe o parâmetro *{id}* e utiliza-o para buscar (no banco) o curso que deve ser removido. Observe que o método *find()* do **Eloquent** é utilizado, assim como no método editar visto anteriormente. O método *delete()*, ao final, efetua a remoção.

Framework Laravel

Exemplo Remove: (execução)

Arquivo "curso.blade.php"



[Home](#)

BD / Migration / Seeder / Model / Controller



Cursos Cadastrados

[Cadastrar Novo Curso](#)

ID	NOME DO CURSO	ABREVIATURA	EVENTOS
1	ANÁLISE E DESENVOLVIMENTO DE SISTEMAS	TADS	Editar Remover
2	EMI EM INFORMÁTICA	INFO	Editar Remover
4	PÓS-GRADUAÇÃO EM MATEMÁTICA COMPUTACIONAL	PGMC	Editar Remover

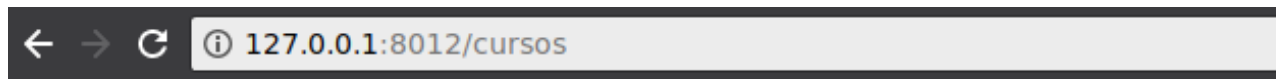
Selecione a opção **Remover**.

©2018 » Gil Eduardo de Andrade

Framework Laravel

Exemplo Remove: (execução)

Arquivo "curso.blade.php"



[Home](#)

BD / Migration / Seeder / Model / Controller



Cursos Cadastrados

[Cadastrar Novo Curso](#)

ID	NOME DO CURSO	ABREVIATURA	EVENTOS
1	ANÁLISE E DESENVOLVIMENTO DE SISTEMAS	TADS	Editar Remover
2	EMI EM INFORMÁTICA	INFO	Editar Remover

©2018 » Gil Eduardo de Andrade

O curso selecionado anteriormente foi removido.



Conceitos Iniciais

Exemplos Utilizados no Documento

http://www.gileduardo.com.br/ifpr/dwii/downloads/dwii_exdoc07.zip

Código-fonte da Aplicação SETA

http://www.gileduardo.com.br/ifpr/dwii/downloads/dwii_dica07.zip

Exercício sobre o Conteúdo

http://www.gileduardo.com.br/ifpr/dwii/downloads/dwii_pratica07.pdf

