

WEBLogic MVC Play Book

© 2016-2017 WebLogic MVC Framework (ArmoredCore release)

Document version v0.5 (for AC.v085)

By *Sílvia Priem Mendes, PhD and Tiago Monteiro.*

Índice

Introdução	2
1. Criar um controlador. Exemplo genérico:	2
2. Criar um action method. Exemplo:	3
3. Criar um modelo (OPCIONAL). Exemplo:	3
4. Usar um modelo ActiveRecord (OPCIONAL). Exemplo:.....	3
5. Devolver uma vista.....	4
6. Exemplo: (Ficheiro index.phtml encontra-se na pasta view/test)	4
7. Definir uma rota (no ficheiro router.php) para um action method do controlador. ..	4
8. Definição de múltiplas rotas (no ficheiro router.php) no respetivo controlador	5
9. Exemplo: (Declaração de um action method no controlador UserController):	6
 API WebObjects por componente MVC	7
1. WebObjects Usados no Controlador:	7
2. WebObjects Usados na Vista (Ficheiros com extensão .phtml):	8
3. Interfaces	9

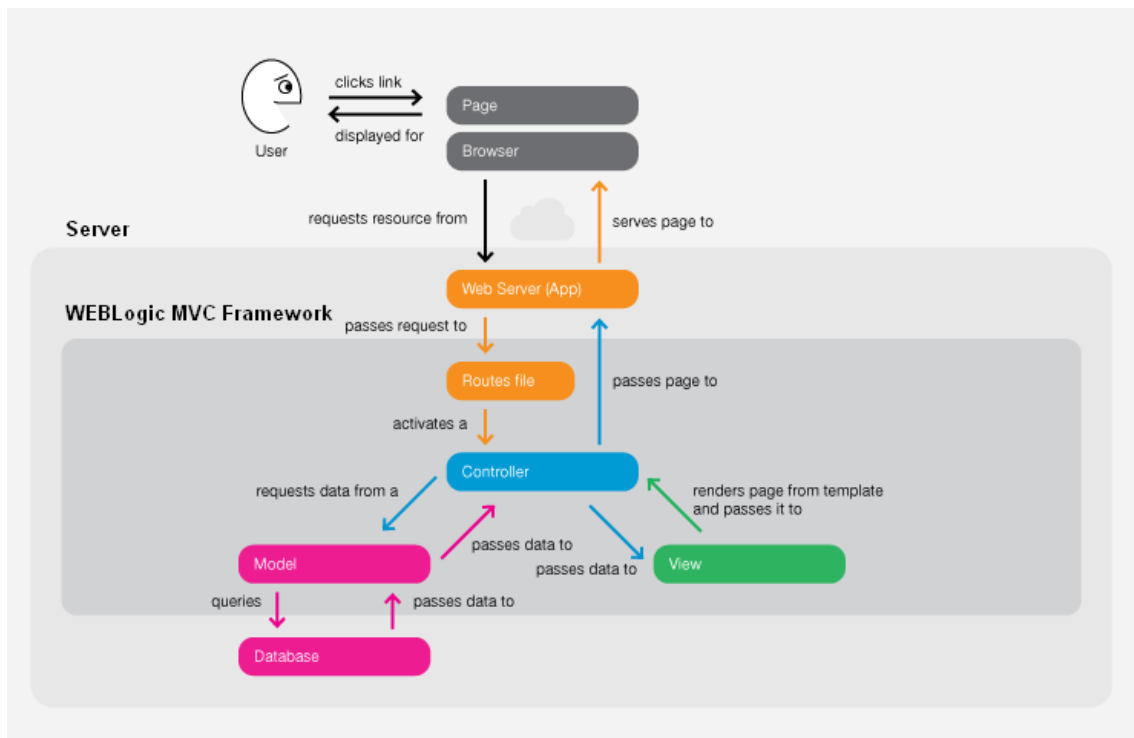


Ilustração 1 - Arquitetura MVC

Introdução

1. Criar um controlador. Exemplo genérico:

```

use ArmoredCore\Controllers\BaseController;

class NomeControlador extends BaseController {

}

```

Um controlador estende sempre de `ArmoredCore\Controllers\BaseController`

Exemplo 2: (Criação do controlador `TestController`. Nome do ficheiro: `TestController.php` na pasta *controller*)

```

use ArmoredCore\Controllers\BaseController;

class TestController extends BaseController {

}

```

2. Criar um action method. Exemplo:

```
class TestController extends BaseController {  
  
    public function showLandingPage() () {  
        // Do Stuff Here  
    }  
  
}
```

3. Criar um modelo (OPCIONAL). Exemplo:

(Criação do ficheiro Test.php na pasta *model*)

```
<?php  
  
class Test {  
  
    public function init_game() {  
        // ADD Logic Init Game Here  
    }  
  
    public function check_end_game() {  
        // Add Login Here  
    }  
  
    // ADD More Functions As Needed  
  
}
```

4. Usar um modelo ActiveRecord (OPCIONAL). Exemplo:

(Criação do ficheiro Book.php na pasta *model*)

```
<?php  
use ActiveRecord\Model;  
  
class Book extends Model {  
  
}
```

A partir deste momento, é possível queries à tabela books da base de dados.
Exemplos (Código realizado no controlador chamado BookController):

```
public function index() {  
    // A seguinte instrução realiza a query (Select * From books)  
    // e guarda o conteúdo numa variável  
    $books = Book::all();  
    ...  
    return View::make('book.index', ['books' => $books]);  
}  
  
public function show($id) {  
    // A seguinte instrução realiza a query  
    //(Select * From books Where id=$id)  
    // e guarda o conteúdo numa variável  
    $book = Book::find($id);  
    ...  
    return View::make('book.show', ['book' => $book]);  
}
```

5. Devolver uma vista

```
public function index() {  
    return View::make('nome_pasta.nome_ficheiro');  
}
```

6. Exemplo: (Ficheiro index.phtml encontra-se na pasta view/test)

```
public function index() {  
    return View::make('test.index');  
}
```

7. Definir uma rota (no ficheiro router.php) para um action method do controlador.

Exemplos:

//Definição de uma rota que responde a um pedido GET e respetivo método no controlador a ser executado

```
Router::get('nome_da_rota', 'NomeControlador/nome_do_metodo');
```

```
Router::get('test/index', 'TestController/index');
```

//Definição de uma rota que responde a um pedido HTTP POST e respetivo método no controlador a ser executado

```
Router::post('test/store', 'TestController/store');
```

8. Definição de múltiplas rotas (no ficheiro router.php) no respetivo controlador

Exemplo:

```
Router::resource('user', 'UserController');
```

Automaticamente os seguintes URL's ficam disponíveis, mas cabe ao utilizador implementar as acções no controlador para produzir o efeito desejado/esperado.

Verb (HTTP)	Resource URL	Action	Effect
GET	/user / index	index	Returns a view with a list of users
GET	/ user / create	create	Returns a view with a form to create a new user
POST	/ user / store	store	Stores data from new user form
GET	/ user / show/ {id}	show	Returns a view with the details of user with PK = id
GET	/ user / edit / {id}	edit	Returns a view with a form to edit a user with PK = id
POST	/ user / update / {id}	update	Stores data from the edit user form
POST	/ user / destroy / {id}	destroy	Deletes a user with PK = id

De notar que o {id} é um identificador numérico que representa o id de um utilizador neste caso. Por exemplo, ao acedermos ao URL (user/show/2), vai ser procurado na tabela users o registo com id=2.

De notar que qualquer outro método declarado no controlador, ficará automaticamente mapeado sem que o utilizador necessite de declarar uma nova rota.

9. Exemplo: (Declaração de um action method no controlador UserController):

```
public function custom_show($id) {  
    View::make('user.custom_show', ['user' => $user]);  
}
```

Assim, no browser é possível aceder ao URL que termine com o sufixo:
/user/custom_show/1

API WebObjects por componente MVC

1. WebObjects Usados no Controlador:

ArmoredCore\WebObjects\View

Por exemplo: `View::make('pasta.vista');`

```
use ArmoredCore\WebObjects\View;

public function index() {
    return View::make('test.index');
}
```

ArmoredCore\WebObjects\Redirect

Por exemplo: `Redirect::toRoute('nome_da_rota');`

`Redirect::flashToRoute('nome_da_rota', 'nome_da_entidade');`

```
use ArmoredCore\WebObjects\Redirect;

public function update($id) {
    if($book->is_valid()){
        $book->save();
        Redirect::toRoute('book/index');
    } else {
        // return form with data and errors
        Redirect::flashToRoute('book/edit', ['book' => $book], $id);
    }
}
```

ArmoredCore\WebObjects\Post

Por exemplo: `Post::get('campo_do_form');` ou `Post::getAll();`

```
use ArmoredCore\WebObjects\Post;

public function store()
{
    $dados = Post::getAll();
    $credito = Post::get('credito');
    ...
}
```

ArmoredCore\WebObjects\Session

Exemplo: Session::set('objeto', \$nome_variavel)

P.S. :A sua utilização não requer session_start()

```
<?php use ArmoredCore\WebObjects\Session; ?>

// Definir objeto na sessão
Session::set('object', $dataObject);

// Devolver objeto na sessão
$res = Session::get('object');

// Destruir a sessão
Session::destroy();
```

2. WebObjects Usados na Vista (Ficheiros com extensão .phtml):

ArmoredCore\WebObjects\Data

Exemplo: Obter uma variável para ser usada na vista

```
<?php use ArmoredCore\WebObjects\Data;
$variavel = Data::get('nome_variavel') ?>
```

ArmoredCore\WebObjects\Layout

Exemplo: Incluir uma vista que possa ser reutilizada.

```
<?php use ArmoredCore\WebObjects\Layout;
Layout::includeLayout('header') ?>
```

ArmoredCore\WebObjects\URL

Exemplo: Link para a rota book/show/{id}

```
<?php use ArmoredCore\WebObjects\URL; ?>
...
<a
    href="<?= Url::toRoute('book/show', $book->id) ?>"
    class="btn btn-info"
    role="button">Show</a>
```

ArmoredCore\WebObjects\Asset

Exemplo: Utilizar um ficheiro javascript (.js) que está dentro da pasta public/js

```
<?php use ArmoredCore\WebObjects\Asset; ?>

<script src="<?= Asset::js('template.js') ?>"></script>
```

Exemplo: Utilizar um ficheiro .css que está dentro da pasta public/css

```
<?php use ArmoredCore\WebObjects\Asset; ?>

<link rel="stylesheet" href="<?= Asset::css('main.css') ?>">
```

Exemplo: Utilizar um ficheiro .png que está dentro da pasta public/img

```
<?php use ArmoredCore\WebObjects\Asset; ?>

<a
    class="navbar-brand"
    href="<?= URL::toRoute('home/index') ?>">
    
</a>
```

3. Interfaces

Exemplo: ResourceControllerInterface

```
<?php use ArmoredCore\Interfaces\ResourceControllerInterface; ?>

class BookController
    extends BaseController
    implements ResourceControllerInterface {

    ...

}
```

Outros:

Na framework é utilizado uma implementação de um ActiveRecord que facilita a interação com bases de dados. Ver link: <http://www.phpactiverecord.org>