



WEB разработка

Laravel CRUD

Работа с потребителски данни

Съдържание

Laravel resource controllers and CRUD

Създаване на resource контролери

Routes&resource контролери

post, get, put, delete etc

route:list, except, only /route/

Security

- CSRF protection

- data escaping

Добавяне на пакети -

- Laravel Collective - HTML&Forms

Задача 1

Отпечатайте всички курсове на страница курсове с бутон за промяна и изтриване. Добавете бутон за добавяне на нов курс..

Отпечатайте списък на всички студенти с бутон за изтриване и бутон към профила им. Добавете бутон за добавяне на нов студент.

*въведете предварително нужната информация в базата данни

Laravel resource controllers

Laravel resource controllers

Resource Controllers

Laravel позволява декларирането на "CRUD" пътищата `/routes/` свързани с даден контролер с един ред код.

Засдача Създайте контролер, обработващ всички HTTP заявки, свързани с лекциите, които се пазят във вашето приложение.

//създаваме контролер

php artisan make:controller CourseController --resource

//командата създава контролера в `app/Http/Controllers/CourseController.php`.

`CourseController` съдържа метод за всяка от CRUD операциите.

Laravel resource controllers

Следваща стъпка - регистриране на resourceful routes за **LectureController** -

Route::resource('lectures', 'LectureController')

С тази декларация регистрираме нужните пътища, които ще направят възможни действията свързани с управляването на съответния ресурс - лекциите в приложението.

Използвайте командата -

php artisan route:list

И разгледайте съществуващите пътища, отнасящи се до **LectureController**.

Laravel resource controllers

Какви заявки
изпълнява
resource
контролера ?

Actions Handled By Resource Controller

Verb	URI	Action	Route Name
GET	/photos	index	photos.index
GET	/photos/create	create	photos.create
POST	/photos	store	photos.store
GET	/photos/{photo}	show	photos.show
GET	/photos/{photo}/edit	edit	photos.edit
PUT/PATCH	/photos/{photo}	update	photos.update
DELETE	/photos/{photo}	destroy	photos.destroy

Laravel resource controllers

Spoofing Form Methods

HTML формите не изпълняват PUT, PATCH, or DELETE заявки.

Добавяме скрито `_method` поле. С този трик формите разбират заявката и я изпълняват.

```
{{ method_field('PUT') }}
```


Laravel resource controllers

Частични Resource пътища

Можем да изключим част от пътищата, които се декларират по подразбиране с `Route::resource`.

Така ограничаваме resource контролера и той изпълнява само част от действията.

```
Route::resource('courses', CourseController, ['only' => ['index', 'show']]);
```

//позволяваме само показването на списъка от всички курсове и показването на един курс.

//Никакви други действия, свързани с курсовете не са достъпни.

```
Route::resource('courses', CourseController, ['except' => ['create', 'store', 'update', 'destroy']]);
```

//не са достъпни действията по създаване, запазване, актуализиране и изтриване

Laravel resource controllers

Даване на имена на Resource Routes

Всички действия/заявки/, които изпълнява resource контролера имат име по подразбиране, което може да бъде променено.

```
Route::resource('courses', 'CoursesController', ['names' => ['create' => 'lecture.build']]);
```

//проверете с route:list как изглежда пътя, регистриран по този начин

Пътищата, чиито имена променят, се добавят като елементи в масива 'names'

```
['names' => ['create' => 'course.build',  
             'show' => 'yourCustomRouteNameHere']
```

Laravel resource controllers

Допълване на Resource Контролерите

Ако е необходимо, може да добавите пътища към основните.

Допълнителните пътища трябва да бъдат дефинирани преди декларацията `Route::resource`.

```
Route::get(courses'/additional', CoursesController@methodName'); //допълнителен
```

```
Route::resource(courses', CoursesController'); //основни
```

CSRF и Ларавел

CSRF и Ларавел

CSRF/Cross-site request forgeries/ се отнасят за случаите, в които неоторизирани заявки се изпълняват от името на оторизиран потребител.

Предпазването от CSRF в Ларавел става с автоматично генериран CSRF "token" за всяка активна потребителска сесия.

С този token се удостоверява, че оторизирания потребител е този, който прави заявка към приложението.

Всяка HTML форма в приложението трябва да съдържа скрито CSRF token поле, чрез което да се валидира заявката.

```
<form method="POST" action="/lectures">  
  {{ csrf_field() }}  
  ...  
</form>
```

Задача 2

Направете CRUD за

- Курсовете
- Студенти - техните профили
 - Промяната на профила да става в страницата за профил
 - Изтриването на профил да става от страницата със списъка на всички студенти

Добавяне на допълнителни пакети в Ларавел

Инсталиране

Стъпка 1 Добавяме допълнителните пакет чрез Composer.

Вариант 1

Променяме `composer.json` - добавяме декларация, с която да бъде добавен `laravelcollective/html` към приложението ни.

```
"require": {  
    "laravelcollective/html": "~5.0"//пакета и необходимата версия  
}
```

Актуализираме Composer в конзолата с командата **composer update**

Инсталиране

Вариант 2 Добавяме допълнителните пакет чрез Composer.

Директно в конзолата с командата

composer require **ИМЕТО-НА-ПАКЕТА**

Командата добавя пакета в composer.json

Инсталиране

Стъпка 2 Добавяме новия **provider** в масива **providers** в **config/app.php**:

```
'providers' => [  
    ...  
    'Collective\Html\HtmlServiceProvider',  
    ...  
],
```

//всеки пакет, обикновено, съдържа инструкция за добавяне към Ларавел

Инсталиране

Стъпка 3 Добавяме синоним/и, с който ще извикваме класа/класовете в масива **aliases** в **config/app.php**:

```
'aliases' => [  
    // ...  
    'Form' => 'Collective\Html\FormFacade',  
    'Html' => 'Collective\Html\HtmlFacade',  
    // ...  
],
```

Генериране на форми чрез
`laravelcollective/html`

форми

Отварящ и затварящ form - тагове

```
{!! Form::open(array('url' => 'foo/bar')) !!}  
//  
{!! Form::close() !!}
```

По подразбиране, тези декларации зареждат POST метод.

Може да зададете друг, чрез - `Form::open(array('url' => 'foo/bar', 'method' => 'put'))`

Обърнете внимание: Laravelcollective/html автоматично добавя

- GET, PUT и DELETE методите, както и
- скрито CSRF token поле

За останалите елементи на формата -

<https://laravelcollective.com/docs/5.0/html#text>

форми

Генериране на форми - <https://laravelcollective.com/docs/5.0/html#text>

{{ }} vs {!! !!}

`{{ }}` vs `{!! !!}`

`{}` - ескейпа съдържанието

`{!! !!}` - не ескейпа съдържанието. Само при гарантиран източник на отпечатваното съдържание!!