**Phần 1 – Setup**

1/Override method tokensMatch để không sử dụng token khi gửi request JSON

Mặc định thì middleware app/Http/Middleware/VerifyCsrfToken.php sẽ ngăn cản chúng ta thực hiện request nếu không sử dụng token nhằm tránh kiểu tấn công CSRF. Do đó chúng ta tạm thời bỏ tính năng này bằng cách ghi đè method tokensMatch mà VerifyCsrfToken.php thừa kế từ BaseVerifier như sau:

protected function tokensMatch($request)

{

if ($request->wantsJson()) {

return true;

}

return parent::tokensMatch($request);

}

Restartserver : php artisan serve

2/example

a/Tạo model và migration file cho article

-php artisan make:model Article –migration

-Mở file migration vừa tạo ở database/migrations/\*create\_articles\_table.php lên và thêm các thuộc tính vào:

public function up()

{

Schema::create('articles', function (Blueprint $table) {

$table->increments('id');

$table->string('title');

$table->text('content');

$table->timestamps();

});

}

- run command php artisan migrate

b/ Thêm danh sách các thuộc tính có thể mass assignment

Mở model app/Article.php ra và thêm mảng $fillable như sau:

class Article extends Model

{

protected $fillable = ['title', 'content'];

}

Thuộc tính $fillable xác định các cột có thể được insert/update bằng mass assignment:

<https://laravel.com/docs/5.2/eloquent#mass-assignment>

c/ Tạo một article trong database sử dụng console

php artisan tinker

App\Article::create(['title' => 'Article Title', 'content' => 'Article Content']);

d/ Tạo controller cho articles

php artisan make:controller ArticlesController

thêm RESTful method trong app/Http/Controllers/ArticlesController.php

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\Http\Requests;

use App\Http\Controllers\Controller;

use App\Article;

class ArticlesController extends Controller

{

public function index()

{

return Article::all();

}

public function store(Request $request)

{

return Article::create($request->all());

}

public function show(Article $article)

{

return $article;

}

public function update(Request $request, Article $article)

{

$article->update($request->all());

return $article;

}

public function destroy(Article $article)

{

return (string) $article->delete();

}

}

e/ Thêm routes cho articles : app/Http/routes.php

Route::singularResourceParameters();

...

Route::resource('articles', 'ArticlesController');

singularResourceParameters để giải quyết vấn đề số ít và số nhiều trong tên của resource

https://github.com/laravel/framework/issues/11644