

# RESTful API with Laravel Build a real API with Laravel

## 1. INTRODUCTION

## 2. DOWNLOADING AND INSTALLING THE TOOLS TO DEVELOP THE RESTFUL API WITH LARAVEL

## 3. MIGRATIONS

## 4. SEEDING (SEED AND FACTORY)

## 5. MODEL

## 6. CONTROLLER (RESOURCES – API RESOURCES)

## 6. RELATIONSHIPS

(ONE TO ONE / ONE TO MANY / MANY TO MANY)

## 7. TRAIT

```
namespace App\Traits;

use Illuminate\Database\Eloquent\Collection;
use Illuminate\Database\Eloquent\Model;
use PhpParser\Node\Expr\AssignOp\Mod;

trait ApiResponse {

    private function successResponse($data, $code)
    {
        return response()->json($data, $code);
    }

    protected function errorResponse($message, $code)
    {
        return response()->json(['error' => $message, 'code' => $code], $code);
    }

    protected function showAll(Collection $collection, $code = 200)
    {
        return $this->successResponse(['data' => $collection], $code);
    }
}
```

## 8.SCOPE

```
app > Scopes > BuyerScope.php > BuyerScope
1  <?php
2  namespace App\Scopes;
3  use Illuminate\Database\Eloquent\Model;
4  use Illuminate\Database\Eloquent\Scope;
5  use Illuminate\Database\Eloquent\Builder;
6  class BuyerScope implements Scope
7  {
8      public function apply(Builder $builder, Model $model)
9      {
10         $builder->has('transactions');
11     }
12 }
13
```

## 9.ROUTE

```
Route::resource('products', 'Product\ProductController', ['only' => ['index', 'show']]);
Route::resource('products.transaction', 'Product\ProductTransactionController', ['only' => ['index', 'show']]);
Route::resource('products.buyer', 'Product\ProductBuyerController', ['only' => ['index', 'show']]);
Route::resource('products.categories', 'Product\ProductCategoryController', ['only' => ['index', 'show']]);

/**
 * Sellers
 */
Route::resource('sellers', 'Seller\SellerController', ['only' => ['index', 'show']]);
Route::resource('sellers.transactions', 'Seller\SellerTransactionController', ['only' => ['index', 'show']]);
Route::resource('sellers.categories', 'Seller\SellerCategoryController', ['only' => ['index', 'show']]);
Route::resource('sellers.products', 'Seller\SellerProductController', ['except' => ['create', 'store']]);

/**
 * Transactions
 */
```

## 10. SOFT DELETING

## 11. HANDLING ERRORS AND EXCEPTIONS WITH THE LARAVEL HANDLER

```
ceptions > Handler.php > NotFoundHttpException
public function render($request, $exception)
{
    if($exception instanceof ValidationException ){
        return $this->convertValidationExceptionToResponse($exception , $request)
    }

    if ($exception instanceof ModelNotFoundException) {
        $modelName = strtolower(class_basename($exception->getModel()));
        return $this->errorResponse("Does not exists any {$modelName} with the sp

    }

    if($exception instanceof AuthenticationException ){
        return $this->unauthenticated($request , $exception);
    }

    if($exception instanceof AuthorizationException ){
        return $this->errorResponse( $exception->getMessage() , 403);
    }
}
```