ALTERAR O DIRETÓRIO PADRÃO DOS MODELS NO LARAVEL

Por default, no Laravel, os models estão localizados dentro do namespace App, o que pode se tornar muito confuso ou até mesmo desorganizado se você estiver trabalhando em uma aplicação grande.  
Decidi então organizar meus models dentro do namespace App\Models alterando esse padrão.

A primeira coisa que você precisa fazer é criar, dentro diretório App um diretório com o nome Models. Em seguida mover o model User.php para a pasta App/Models (o diretório recém criado) e atualizar o namespace de acordo.

Isso requer que você atualize todos os arquivos que referenciam a classe App\User. Vamos à eles;

O primeiro é o próprio User.php. Altere o namespace App; de;

namespace App;

para;

namespace App\Models;

O segundo é o auth.php dentro do diretório config de;

'providers' => [

'users' => [

'driver' => 'eloquent',

'model' => App\User::class,

],

para;

'providers' => [

'users' => [

'driver' => 'eloquent',

'model' => App\Models\User::class,

],

O terceiro é o controller RegisterController.php dentro do diretório App\Http\Controllers\Auth de;

<?php

namespace App\Http\Controllers\Auth;

use App\User;

use App\Http\Controllers\Controller;

.....

para;

<?php

namespace App\Http\Controllers\Auth;

use App\Models\User;

use App\Http\Controllers\Controller;

.....

Depois o arquivo services.php dentro do diretório config de;

'stripe' => [

'model' => App\User::class,

'key' => env('STRIPE\_KEY'),

'secret' => env('STRIPE\_SECRET'),

],

para;

'stripe' => [

'model' => App\Models\User::class,

'key' => env('STRIPE\_KEY'),

'secret' => env('STRIPE\_SECRET'),

],

Por fim, mas não menos importante, o arquivo UserFactory.php dentro do diretório database/factories de ;

$factory->define(App\User::class, function (Faker $faker) {

...

});

para;

$factory->define(App\Models\User::class, function (Faker $faker) {

...

});

No terminal digite composer dumpautoloade clique “enter”.

Legal! Nós mudamos o namespace do nosso Models\User, mas o que dizer de gerar novos models?  
Como sabemos, eles são colocados dentro do namespace App por padrão.

Para corrigir isso, podemos estender o padrão ModelMakeCommand;

Agora vamos sobrescrever o caminho do diretório no comando *Artisan* make:model;

No terminal digite php artisan make:command ModelMakeCommand

Dentro do diretório App\Console\Commands abra o arquivo ModelMakeCommand.php, apague seu conteúdo e digite (ou copie e cole) o seguinte código;

<?php

namespace App\Console\Commands;

use Illuminate\Foundation\Console\ModelMakeCommand as Command;

class ModelMakeCommand extends Command

{

/\*\*

\* Get the default namespace for the class.

\*

\* @param string $rootNamespace

\* @return string

\*/

protected function getDefaultNamespace($rootNamespace)

{

return "{$rootNamespace}\Models";

}

}

Por fim registre-o no Kernel;

Dentro do diretório App\Console abra o arquivo kernel.php, encontre o protected $commands e digite (ou copie e cole) o seguinte código;

/\*\*

\* The Artisan commands provided by your application.

\*

\* @var array

\*/

protected $commands = [

Commands\ModelMakeCommand::class

];

No terminal digite composer dumpautoloade clique “enter”.

Depois digite php artisan make:model NomeDaModele clique “enter”. Veja o resultado;

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Model;

class NomeDaModel extends Model

{

//

}

Agora, quando você executar o comando php artisan make:model NomeDaModel, ele assumirá como padrão o diretório app\Models\NomeDaModel.php. E se você quiser usar parâmetros como;  
php artisan make:model NomeDaModel -mcr (para criar o model junto com a migration e o controller com os resources) funcionará perfeitamente.