

## Bài 10: Hướng dẫn tách riêng file route trong Laravel

Trong các dự án Laravel, khi ứng dụng phát triển lớn dần, file routes/web.php dễ bị dài và khó quản lý. Giải pháp là tách các nhóm route theo chức năng sang các file riêng, ví dụ routes/news.php.

### 1. Tạo file route mới

Trong thư mục routes/, tạo file mới **news.php** với nội dung như sau:

```
<?php

use Illuminate\Support\Facades\Route;
use App\Http\Controllers\NewsController;

//move các route liên quan đến news chuyển hết vào đây
Route::resource('news', NewsController::class);
```

---

### 2. Đăng ký file route trong RouteServiceProvider

Với Laravel 10 trở về trước

Mở file app/Providers/**RouteServiceProvider.php** và chỉnh sửa phương thức boot():

```
public function boot()
{
    $this->routes(function () {
        Route::middleware('web')
            ->group(base_path('routes/web.php'));

        // Thêm file news.php
        Route::middleware('web')
            ->group(base_path('routes/news.php'));
    });
}
```

---

Với Laravel 11, 12 trở về sau:

Laravel không còn viết sẵn **RouteServiceProvider** với phương thức **boot()** nữa, mà việc load routes được định nghĩa trực tiếp trong file **routes/web.php** và **routes/api.php**. Do đó bạn sẽ không thấy RouteServiceProvider.php như tài liệu các bản cũ.

#### 1. Cách đơn giản nhất: include trực tiếp trong routes/web.php

Mở routes/web.php, thêm dòng:

```
require __DIR__ . '/news.php';
```

---

Laravel sẽ load thêm file `routes/news.php` cùng với các route mặc định.

---

## 2. Nếu bạn vẫn muốn tách logic kiểu cũ (tự tạo Service Provider)

Bạn có thể tạo thủ công `RouteServiceProvider` bằng lệnh:

```
php artisan make:provider RouteServiceProvider
```

---

Rồi trong đó tự đăng ký các file route bổ sung.

## 3. Cách chính thống — đăng ký trong bootstrap/app.php (recommended when you want centralized route registration)

Mở `bootstrap/app.php`. Laravel 11/12 cho phép cấu hình routing tại đây bằng `withRouting()` và truyền thêm một closure (**then**) để đăng ký route files bổ sung.

```
<?php
use Illuminate\Foundation\Application;
use Illuminate\Support\Facades\Route;

return Application::configure(basePath: dirname(__DIR__))
    ->withRouting(
        web: __DIR__.'/../routes/web.php',
        api: __DIR__.'/../routes/api.php',
        commands: __DIR__.'/../routes/console.php',
        // optionally health, apiPrefix, ...
        then: function () {
            // đăng ký đơn giản file routes/news.php dưới middleware 'web'
            Route::middleware('web')->group(base_path('routes/news.php'));

            // hoặc đăng ký file admin với prefix/name:
            //bổ sung thêm tiền tố admin cho các uri của route news
            // Route::middleware('web')
            //     ->prefix('admin')
            //     ->name('admin.')
            //     ->group(base_path('routes/news.php'));
        },
    )->create();
```

Nhưng thực tế từ Laravel 11+, cách **require file trong web.php** (cách 1) là khuyến nghị, đơn giản và đủ dùng.

### 3. Kiểm tra hoạt động

Chạy lệnh kiểm tra route:

**php artisan route:list**

---

Chèn hình ảnh kết quả vào đây

### 4. Ưu điểm

- Dễ quản lý các route theo module (news, client, admin...).
- Dễ bảo trì, tránh file web.php quá dài.
- Có thể kết hợp với prefix và middleware trong từng file route riêng.