

Course - Laravel Framework

Xác nhận Email

Nhiều ứng dụng web yêu cầu người dùng xác minh địa chỉ email của họ trước khi sử dụng ứng dụng. Thay vì buộc bạn phải triển khai lại tính năng này bằng tay cho mỗi ứng dụng bạn tạo, Laravel cung cấp các service tích hợp tiện lợi để gửi và xác minh các yêu cầu xác minh email.

Tags: email verification, xác nhận email, laravel

Giới thiệu

Nhiều ứng dụng web yêu cầu người dùng xác minh địa chỉ email của họ trước khi sử dụng ứng dụng. Thay vì buộc bạn phải triển khai lại tính năng này bằng tay cho mỗi ứng dụng bạn tạo, Laravel cung cấp các service tích hợp tiện lợi để gửi và xác minh các yêu cầu xác minh email.

Bạn muốn bắt đầu nhanh? Cài đặt một trong các bộ khởi động nhanh ứng dụng Laravel trong một ứng dụng Laravel mới. Bộ dụng cụ khởi động này sẽ thực hiện toàn bộ hệ thống xác thực của bạn, bao gồm cả hỗ trợ xác minh email.

Việc chuẩn bị về model

Trước khi bắt đầu, hãy xác minh rằng model `App\Models\User` của bạn triển khai contract `Illuminate\Contracts\Auth\MustVerifyEmail`:

```
<?php

namespace App\Models;

use Illuminate\Contracts\Auth\MustVerifyEmail;
use Illuminate\Foundation\Auth\User as Authenticatable;
use Illuminate\Notifications\Notifiable;

class User extends Authenticatable implements MustVerifyEmail
{
    use Notifiable;

    // ...
}
```

Khi interface này đã được thêm vào model của bạn, những người dùng mới đăng ký sẽ tự động được gửi một email có chứa liên kết xác minh email. Như bạn có thể thấy bằng cách kiểm tra `App\Providers\EventServiceProvider` của ứng dụng, Laravel đã chứa một chương trình theo dõi `SendEmailVerificationNotification` được đính kèm với event `Illuminate\Auth\Events\Registered`. Chương trình theo dõi event này sẽ gửi liên kết xác minh email cho người dùng.

Nếu bạn đang triển khai thủ công việc đăng ký trong ứng dụng của mình thay vì sử dụng bộ khởi động, bạn nên đảm bảo rằng bạn đang gửi event `Illuminate\Auth\Events\Registered` sau đợt đăng ký thành công của người dùng:

```
use Illuminate\Auth\Events\Registered;  
  
event(new Registered($user));
```

Việc chuẩn bị về cơ sở dữ liệu

Tiếp theo, bảng người dùng của bạn phải chứa cột `email_verified_at` để lưu trữ ngày và giờ địa chỉ email của người dùng được xác minh. Theo mặc định, việc thiết lập bảng user luôn có sẵn trong framework Laravel đã bao gồm cột này. Vì vậy, tất cả những gì bạn cần làm là chạy quá trình thiết lập cơ sở dữ liệu của mình:

```
php artisan migrate
```

Phân tuyến

Để thực hiện xác minh email đúng cách, cần phải khai báo ba route. Đầu tiên, cần có một route hiển thị thông báo cho người dùng rằng họ nên nhấp vào liên kết xác minh email trong email xác minh mà Laravel đã gửi cho họ sau khi đăng ký.

Thứ hai, cần có một route xử lý các request được tạo ra khi người dùng nhấp vào liên kết xác minh email trong email.

Thứ ba, sẽ cần một route gửi lại liên kết xác minh nếu người dùng vô tình làm mất liên kết xác minh đầu tiên.

Mail thông báo xác minh email

Như đã đề cập trước đây, một route sẽ được khai báo để trả về một view hiển thị hướng dẫn người dùng nhấp vào liên kết xác minh email đã được Laravel gửi qua email cho họ sau khi đăng ký. View hiển thị này sẽ được hiển thị cho người dùng khi họ cố gắng truy cập các phần khác của ứng dụng mà không xác minh địa chỉ email của họ trước. Hãy nhớ rằng, liên kết sẽ tự động được gửi qua email cho người dùng miễn là model `App\Models\User` của bạn triển khai từ interface `MustVerifyEmail`:

```
Route::get('/email/verify', function () {  
    return view('auth.verify-email');  
})->middleware('auth')->name('verification.notice');
```

Route trả về thông báo xác minh email phải được đặt tên là **verification.notice**. Điều quan trọng là route được chỉ định với tên này là vì middleware xác minh đi kèm với Laravel sẽ tự động chuyển hướng đến tên route này nếu người dùng chưa xác minh địa chỉ email của họ.

Khi triển khai xác minh email theo cách thủ công, bạn được yêu cầu tự tạo nội dung của view hiển thị thông báo xác minh. Nếu bạn muốn khung sườn bao gồm tất cả các view hiển thị xác thực và xác minh cần thiết, thì hãy xem bộ công cụ khởi động nhanh của ứng dụng Laravel.

Xử lý email xác minh

Tiếp theo, chúng ta cần xác định một route sẽ xử lý các request được tạo ra khi người dùng nhấp vào liên kết xác minh email đã được gửi qua email cho họ. Route này phải được đặt tên là **verification.verify** và được chỉ định xác thực và middleware đã ký:

```
use Illuminate\Foundation\Auth\EmailVerificationRequest;  
  
Route::get('/email/verify/{id}/{hash}', function (EmailVerificationRequest $request) {  
    $request->fulfill();  
  
    return redirect('/home');  
})->middleware(['auth', 'signed'])->name('verification.verify');
```

Trước khi tiếp tục, chúng ta hãy xem xét kỹ hơn route này. Đầu tiên, bạn sẽ nhận thấy rằng chúng ta đang sử dụng loại yêu cầu **EmailVerificationRequest** thay cho đối tượng **Illuminate\Http\Request** điển hình. **EmailVerificationRequest** là một yêu cầu biểu mẫu được bao gồm trong Laravel. Yêu cầu này sẽ tự động xử lý việc xác thực thông số **id** và **hash** của yêu cầu.

Tiếp theo, chúng ta có thể tiến hành gọi trực tiếp phương thức **fulfill** theo yêu cầu. Phương thức này sẽ gọi phương thức **markEmailAsVerified** trên người dùng được xác thực và gửi sự kiện **Illuminate\Auth\Events\Verified**. Phương thức

`markEmailAsVerified` có sẵn cho mô hình `App\Models\User` mặc định thông qua lớp cơ sở `Illuminate\Foundation\Auth\User`. Khi địa chỉ email của người dùng đã được xác minh, bạn có thể chuyển hướng họ đến bất cứ nơi nào bạn muốn.

Gửi lại email xác nhận

Đôi khi người dùng có thể đặt nhầm chỗ hoặc vô tình xóa email xác minh địa chỉ email. Để đáp ứng điều này, bạn có thể muốn tạo một route để cho phép người dùng yêu cầu gửi lại email xác minh. Sau đó, bạn có thể đưa ra yêu cầu đối với route này bằng cách đặt một nút gửi form biểu mẫu đơn giản trong view hiển thị thông báo xác minh của bạn:

```
use Illuminate\Http\Request;

Route::post('/email/verification-notification', function (Request $request) {
    $request->user()->sendEmailVerificationNotification();

    return back()->with('message', 'Verification link sent!');
})->middleware(['auth', 'throttle:6,1']->name('verification.send');
```

Route nội bộ

Middleware của route có thể được sử dụng để chỉ cho phép những người dùng đã được xác minh truy cập vào một route nào. Laravel cung cấp middleware đã được xác minh, tham chiếu đến class `Illuminate\Auth\Middleware\EnsureEmailIsVerified`. Vì middleware này đã được đăng ký trong hạt nhân HTTP kernel của ứng dụng của bạn, tất cả những gì bạn cần làm là đính kèm middleware vào phần khai báo route:

```
Route::get('/profile', function () {
    // Only verified users may access this route...
})->middleware('verified');
```

Nếu một người dùng chưa được xác minh cố gắng truy cập vào một route đã được chỉ định middleware này, họ sẽ tự động được chuyển hướng đến route có tên là `verification.notice`.

Tự tạo

Tùy biến email xác minh

Mặc dù mail thông báo xác minh email mặc định phải đáp ứng các yêu cầu của hầu hết các ứng dụng, Laravel cho phép bạn tùy chỉnh cách cấu tạo thư thông báo xác minh email.

Để bắt đầu, hãy truyền một hàm xử lý cho phương thức `toMailUsing` được cung cấp bởi thông báo `Illuminate\Auth\Notifications\VerifyEmail`. Hàm xử lý này sẽ nhận được đối tượng model thông báo đang nhận thông báo cũng như URL xác minh email đã đánh dấu mà người dùng phải truy cập để xác minh địa chỉ email của họ. Hàm xử lý sẽ trả về một đối tượng của `Illuminate\Notifications\Messages\MailMessage`. Thông thường, bạn nên gọi phương thức `toMailUsing` từ phương thức boot của lớp `App\Providers\AuthServiceProvider` của ứng dụng:

```
use Illuminate\Auth\Notifications\VerifyEmail;
use Illuminate\Notifications\Messages\MailMessage;

/**
 * Register any authentication / authorization services.
 *
 * @return void
 */
public function boot()
{
    // ...

    VerifyEmail::toMailUsing(function ($notifiable, $url) {
        return (new MailMessage)
            ->subject('Verify Email Address')
            ->line('Click the button below to verify your email address.')
            ->action('Verify Email Address', $url);
    });
}
```

Để tìm hiểu thêm về thông báo qua thư, vui lòng tham khảo tài liệu thông báo qua thư.

Events

Khi sử dụng bộ công cụ khởi động nhanh ứng dụng Laravel, Laravel gửi các event trong quá trình xác minh email. Nếu bạn đang xử lý thủ công xác minh email cho ứng dụng của mình, bạn có thể muốn gửi các event này theo cách thủ công sau khi quá trình xác minh hoàn tất. Bạn có thể đính kèm chương trình theo dõi vào các sự kiện này trong **EventServiceProvider** của ứng dụng của bạn:

```
/**
 * The event listener mappings for the application.
 *
 * @var array
 */
protected $listen = [
    'Illuminate\Auth\Events\Verified' => [
        'App\Listeners\LogVerifiedUser',
    ],
];
```