**MỞ ĐẦU**

Ngày nay khi các lập trình viên cần xây dựng các website và các ứng dụng web phức tạp, họ có thể sẽ mất quá nhiều thời gian và rắc rối nếu cứ xây dựng ứng dụng từ đầu, vì thế cần đến một cách tự nhiên hơn để xây dựng sản phẩm. Chính vì vậy các Web framework ra đời và cung cấp cho các nhà phát triển với một giải pháp thỏa đáng cho điều đó.

Trong thuật ngữ thiết kế web, framework có thể xem như một “khung nền”, một thư viện tổng hợp các lớp đã được xây dựng hoàn chỉnh hay một thư viện chứa nhiều công nghệ của nhiều lĩnh vực hay là bộ khung, sườn để phát triển các ứng dụng. Các lập trình viên, nhà phát triển web sẽ sử dụng các framework này như nguồn nguyên liệu để làm nên các trang web hoàn chỉnh.

[Framework PHP](https://www.webico.vn/framework-la-gi-top-10-framework-php-tot-nhat-danh-cho-lap-trinh-vien/) là một trong các loại framework phổ biến nhất được dùng để thiết kế web. Tuy là phổ biến nhất nhưng nhiều sinh sinh vẫn chưa biết đến Laravel, chính vì vậy nhóm đã chọn đề tài giới thiệu Laravel FrameWork. Bao gồm các nội dung:

**Chương 1:** Giới thiệu tổng quan

**Chương 2:** Những khác biệt so với code thuần

**Chương 3:** Ưu nhược điểm của Laravel FrameWork

**Chương 4:** Kết luận

# CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN

## **1.1. FrameWork là gì?**

**Framework** được định nghĩa là một bộ khung hay giàn giáo cung cấp các chức năng, giải pháp được cài đặt sẵn giúp tiết kiệm thời gian trong quá trình phát triển ứng dụng. Có thể hiểu Framework tương tự như khung của một ngôi nhà nó cung cấp một nền tảng chắc chắn và cơ bản để bạn có thể xây dựng được đa số các kiểu nhà được thiết kế trong bản vẻ. Mặc dù bạn có thể xây hoàn chỉnh một ngôi nhà từ đầu, nhưng bắt đầu với một dàn khung sẽ giúp bạn tiết kiệm được tối đa công sức cho việc bắt đầu.

**Framework trong lập trình ứng dụng** cũng vậy, nó cung cấp đa số các kiểu mẫu thiết kế phù hợp với **ứng dụng** bạn sắp thực hiện, các thư viện, API, trình biên dịch và hơn thế nữa để hợp lý hóa việc tạo ra phần mềm mới. Biết được framework nào để sử dụng cho dự án nào là một kỹ năng quan trọng đối với bất kỳ nhà phát triển nào.

Trong mỗi một ngôn ngữ, một lĩnh vực đều có nhiều các framework được tạo ra bởi các nhóm lập trình hoặc thậm chí là bởi các công ty lớn, vì vậy việc sử dụng lại các framework sẽ có rất nhiều ưu điểm:

+ Framework xây dựng sẵn các tính năng chung, ví dụ dự án web nào cũng có cần có phần quản lý người dùng như đăng ký, đăng nhập, kết nối cơ sở dữ liệu…

+ Sử dụng framework giảm thiểu tối đa thời gian và công sức phát triển ứng dụng.

+ Cho phép ứng dụng kế thừa một cấu trúc được quy định sẵn, sẽ dễ dàng trong vận hành và bảo trì sau này.

Bên cạnh đó sử dụng framework cũng có một số nhược điểm:

+ Phải mất nhiều thời gian để học và làm quen một framework.

+ Kích cỡ ứng dụng khi sử dụng framework là rất lớn, một website có thể lên đến hàng vài trăm MB code khi chưa có bất kỳ nội dung nào.

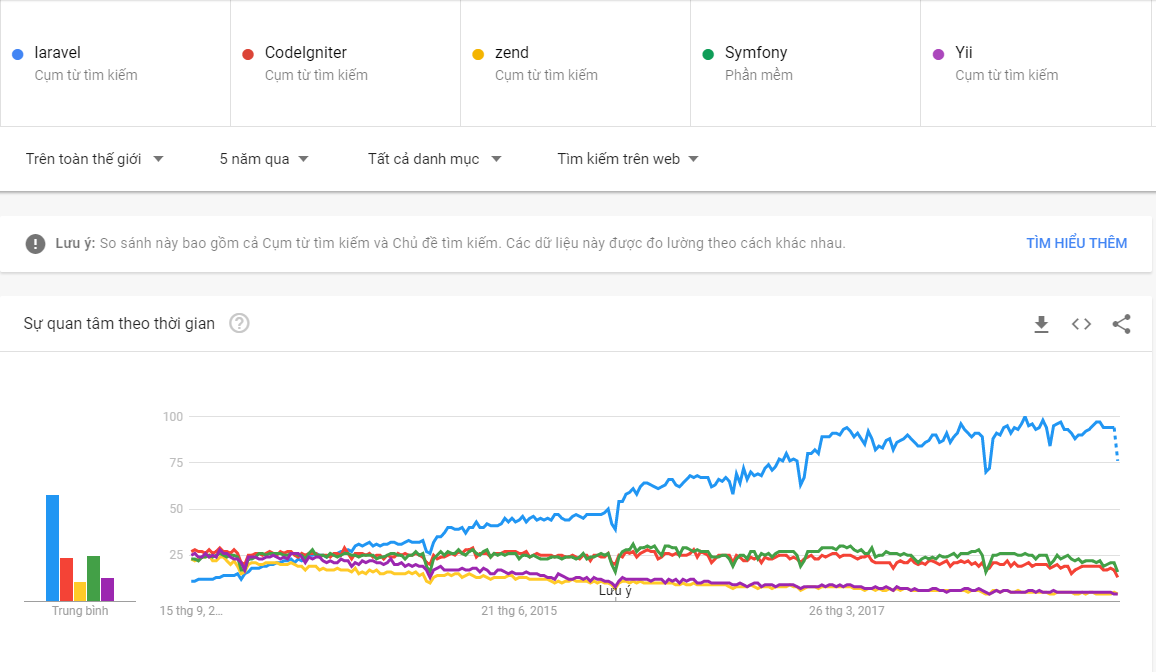
+ Cần viết code tuân thủ theo các tiêu chuẩn chung mà framework đặt ra.

+ Không thích hợp với các ứng dụng quá nhỏ.

## **1.2. Laravel FrameWork là gì?**

Laravel là một framework của ngôn ngữ PHP phổ biến nhất nhất hiện nay, mặc dù framework này mới được ra mắt muộn hơn các PHP framework khác vào tháng 06/2011 bởi Taylor Otwell nhưng Laravel nhanh chóng trở một framework quan trọng và phổ biến nhất của PHP vì nó cung cấp nhiều tính năng hữu ích cho quá trình phát triển nhanh một trang web.

Laravel hỗ trợ nhiều cơ sở dữ liệu khác nhau, cung cấp một hệ sinh thái rộng lớn. Những lập trình viên mới có thể tìm hiểu về Laravel một cách nhanh chóng. Laravel tái sử dụng và sắp xếp các thành phần hiện có để cung cấp cho bạn một framework để xậy dựng các ứng dụng web một cách có cấu trúc và rất thực tế. Nó kết hợp được nhiều ý tưởng tốt từ các ngôn ngữ lập trình cũng như các framework có tiếng như CodeIgniter, Yii, ASP.NET MVC, Ruby on Rails…



(So sánh mức độ phổ biến của Laravel theo Google Trend)

## **1.3. Lý do Laravel FrameWork phổ biến**

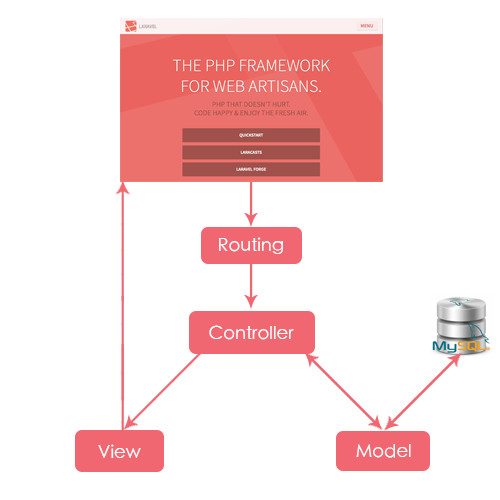
**1.3.1. Dễ dàng sử dụng**

Lý do đầu tiên khiến **Laravel** nhanh chóng được cộng đồng đón nhận và sử dụng nhiều là do nó rất dễ để có thể sử dụng. Ngay cả khi bạn chỉ mới chỉ có những kiến thức cơ bản nhất về lập trình web với PHP thì việc bắt đầu sử dụng **Laravel** cũng chỉ mất vài giờ là bạn có thể bắt tay vào việc làm một project nhỏ.

Document mà **Laravel** cung cấp trên [trang chủ](https://laravel.com/docs/5.5)của mình được viết rất rõ ràng và dể hiểu giúp cho bạn nhanh chóng có thể tìm được những gì mình muốn.

**1.3.2. Xây dựng theo mô hình MVC**

**Laravel** được xây dựng và phát triển theo mô hình MVC (Model-View-Controller) nhờ đó mà cấu trúc và cách tổ chức code trong project được sắp xếp một cách hợp lý dễ dàng cho việc maintain cũng như phát triển về lâu dài.



**1.3.3. Các tính năng được dựng sẵn**

Bản thân **Laravel** đã cung cấp cho người dùng rất nhiều các nhóm tính năng giúp quá trình phát triển trở nên nhanh chóng hơn rất nhiều lần.

Chỉ với các câu lệnh đơn giản: là bạn đã dựng được cho mình toàn bộ các tính năng như đăng nhập, đăng xuất, đăng kí, quên mật khẩu mà không tốn công code một dòng nào cả. Tất nhiên bạn cũng có thể tự mình chỉnh sửa lại logic sao cho hợp với nhu cầu sử dụng cá nhân.

**1.3.4. Cộng đồng hỗ trợ mạnh mẽ**

Chắc hẳn trong quá trình làm việc, bạn có thể sẽ gặp rất nhiều những vấn đề, nhưng bug phát sinh nhưng chưa tìm được câu trả lời. Nhưng nếu bạn sử dụng **Laravel** vướng mắc của bạn có thể đã được người khác giải quyết và bạn có thể lập tức sử dụng đáp án đó hoặc nếu không khi bạn đặt câu hỏi trên các diễn đàn thì cộng đồng đông đảo người sử dụng **Laravel** sẽ hỗ trợ bản giải quyết vấn đề đó.

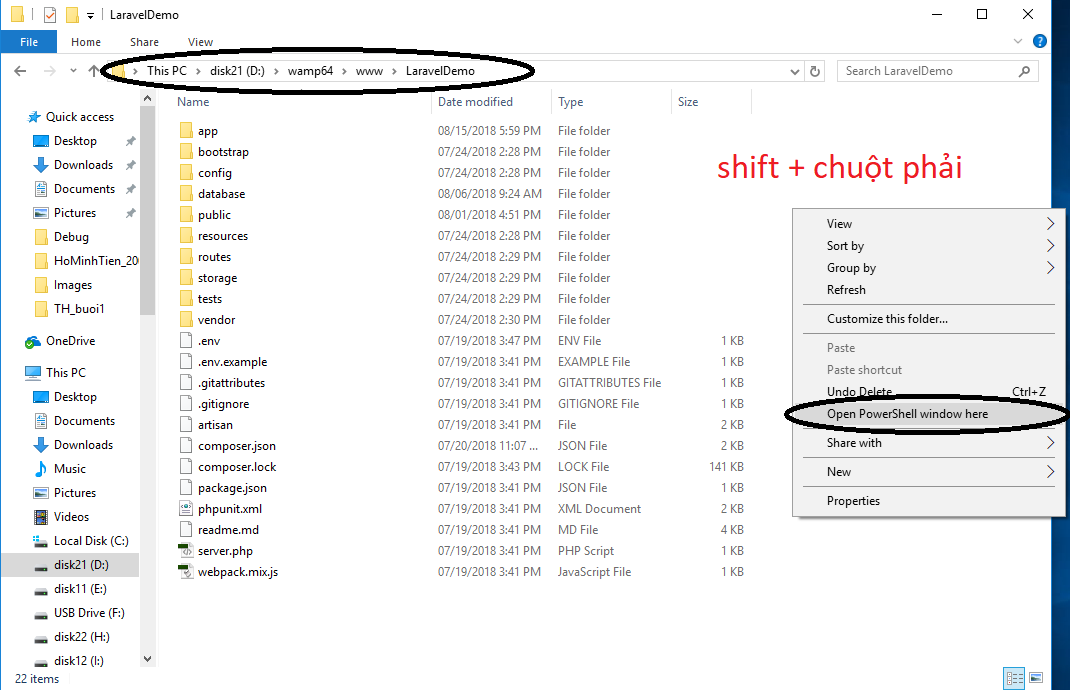
# CHƯƠNG 2: NHỮNG KHÁC BIỆT SO VỚI CODE THUẦN

## **2.1. Tạo project**

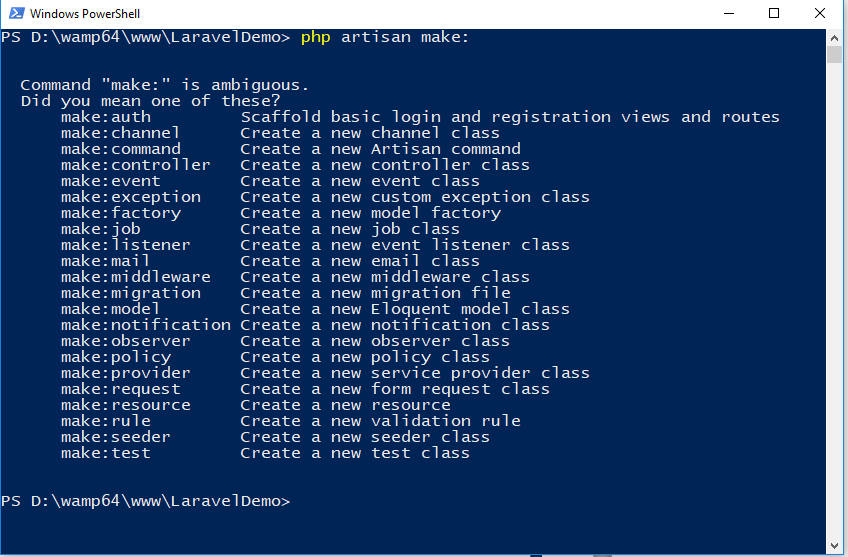
Đối với một project web php thông thường, việc tạo sẽ trở nên rất đơn giản, chỉ cần tạo mới các folder và các file với phần đuôi mở rộng .php ở bất kỳ đâu mà bạn thích. Đối với Laravel thì khác, người dùng vẫn có thể tạo file bằng cách thủ công như code thuần nhưng việc sử dụng framework đòi hỏi người dùng phải nhớ và thêm chính xác các thư viện cần thiết vì thế mà việc tự tạo file sẽ trở nên khó khăn. Biết trước điều đó, Laravel có hỗ trợ *Artisan* giúp người dùng tạo ra một project với các file có hoàn toàn đầy đủ thư viện. Vậy Artisan là gì?

**Laravel Artisan** là một công cụ dòng lệnh được tích hợp sẵn trong các dự án sử dụng Laravel, nó cung cấp rất nhiều các chức năng trợ giúp việc xây dự án, giảm thời gian viết code cũng như tự động hóa một số công việc. Laravel Artisan cung cấp các lệnh cần thiết để hổ trợ quản lý các route, database, controller,...

Để sử dụng Artisan bạn vào thư mục cần lưu file sau đó nhấn *shift + chuột* phải và chọn *Open PowerShell window here* để mở ra giao diện command.

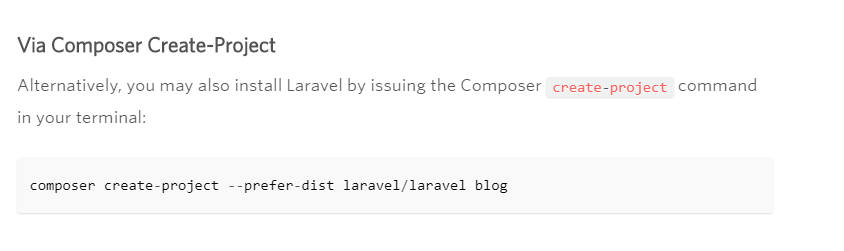


Sau khi mở giao diện command, bạn có thể tạo file bằng cách gõ “php artisan make:Tên\_chức\_năng Tên\_file”. Các chức năng trong artisan bao gồm: auth, model...

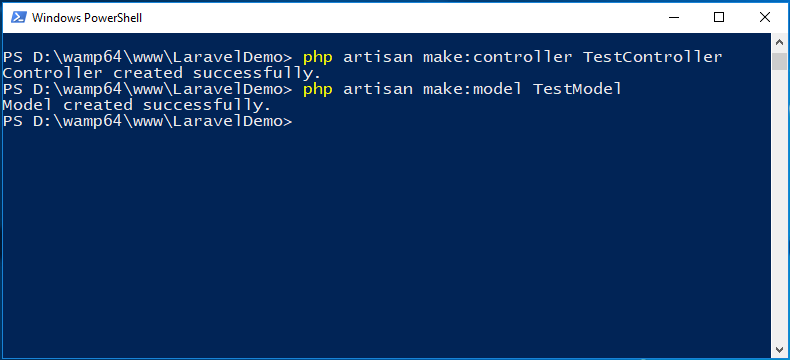


Nhưng để tạo được các file chức năng đó, trước tiên bạn cần phải tạo một project Laravel mới. Để tạo project phiên bản Laravel mới nhất ta vào trang chủ của Laravel là laravel.com chọn Documentation và kéo xuống “Via Composer Create-Project” như

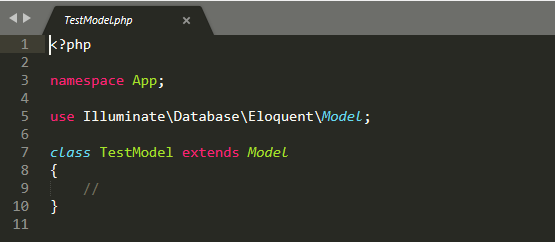
Hình bên dưới và copy lệnh tạo project, sau đó vào command và thực thi lệnh ấy. Việc tạo project cần phải vào trang chủ của laravel để đám bảo rằng bạn sẽ được tạo project với phiên bản Laravel mới nhất.



Sau khi đã có project thì các file chức năng sẽ dễ dàng được tao. Ví dụ để tạo 1 file controller có tên là TestController và 1 Model có tên TestModel ta làm như sau:



Sau khi tạo thành công file sẽ có dạng như sau trong thư muc

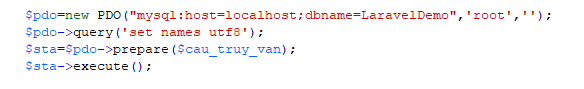


Như vậy, có thể thấy vấn đề đòi hỏi người dùng cần phải nhớ tên thư viện đã được giải quyết một cách nhanh chóng.

## **2.2. Kết nối database**

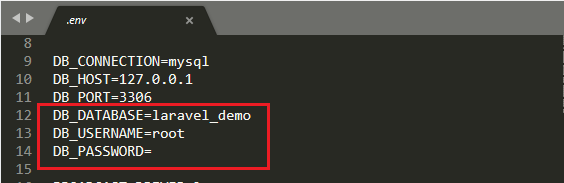
Database là một phần không thể thiếu không chỉ với website mà là với tất cả các ứng dụng. Kết nối database là việc làm hầu như phải có đối với một hệ thống website, thế nhưng quá trình kết nối đòi hỏi người lập trình phải biết rõ từng driver, từng dòng lệnh cấu hình của hệ quản trị cơ sở dữ liệu mà mình muốn kết nối, chưa kể đến việc hiện nay không chỉ có một mà là rất nhiều hệ quản trị cơ sở dữ liệu khác nhau có thể kết nối được với hệ thống website nên để nhớ hết tất cả câu lệnh và driver là rất khó khăn. Hiểu được vấn đề đó, Laravel cũng đã hổ trợ lập trình viên các công cụ cấu hình có sẵn trong nó nhằm giảm tải áp lực cho các nhà phát triển.

Để thấy được rõ nhất vấn đề trên, hãy xem cách để kết nối cơ sở dữ liệu mysql với ngôn ngữ lập trình PHP



Để kết nối mysql với PHP, bạn cần phải thực hiện qua nhiều bước như hình thì mới có thể kết nối được.

Đối với Laravel, tất cả đã được định nghĩa và chuẩn bị sẵn tất cả, bạn chỉ cần điền một số thông tin cần thiết vào là có thể kết nối.



Kết nối myslq với Laravel đã trở nên đơn giản hơn nhiều, bạn cần điền và tên database, mật khẩu và tên đăng nhập là quá trình kết nối đã hoàn tất.

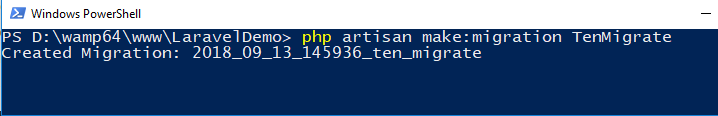
## **2.3. Quản lý database**

Tương tự như cách kết nối, Laravel cũng hỗ trợ tối đa cho lập trình viên trong công cuộc quản lý database trong quá trình sử dụng. Cụ thể là hai tính năng Migrate và Seed.

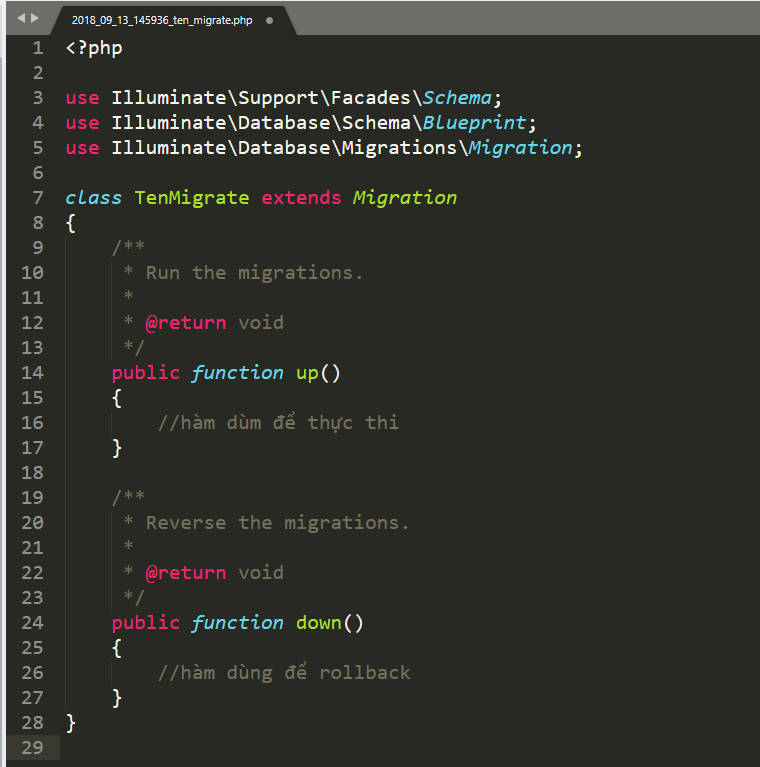
**2.3.1. Migrate**

Migrate dùng để tạo lên cấu trúc bảng trong database, nó giống như một control database có tác dụng quản lý cũng như lưu trữ lại cấu trúc của database giúp cho việc sửa đổi database trở lên dễ dàng hơn.

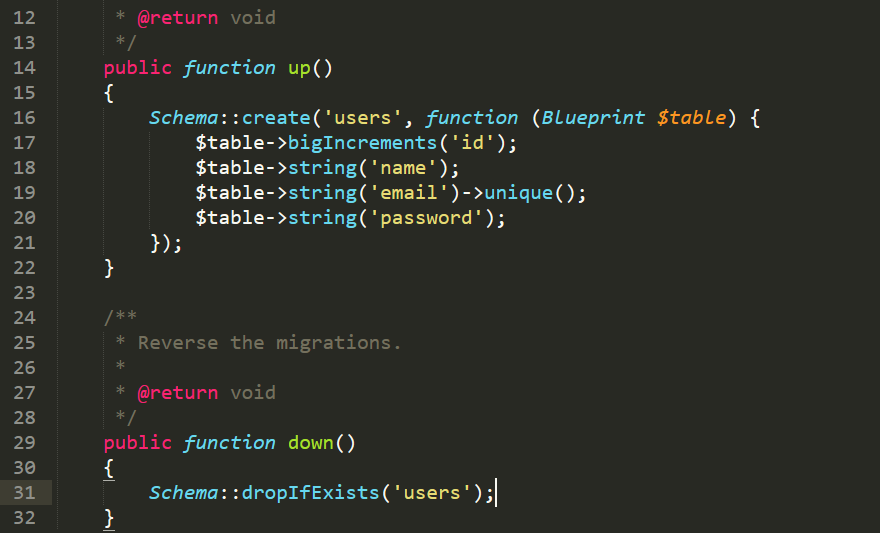
Để tạo ra migrate chỉ cần sử dụng artian với câu lện như bên dưới:



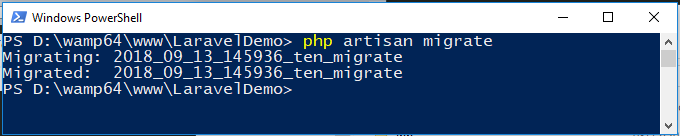
File migare được tạo ra sẽ có dạng như sau:



Function up() thường là dùng để tạo bảng, function down() thường có chức năng dùng để rollback lại trang thái ban đầu tại thời điểm mà function up() chưa được thực thi. Chức năng của function down() thường là xóa bảng (drop table).Ví dụ: để tạo ra một table có tên users ta làm như sau:



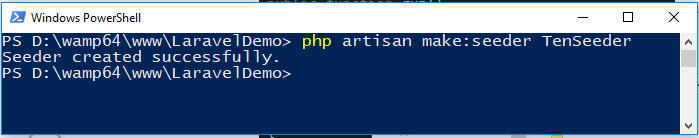
Việc quan trong cuối cùng trong bước tạo table bằng migrate là thực thi nó bằng artisan:



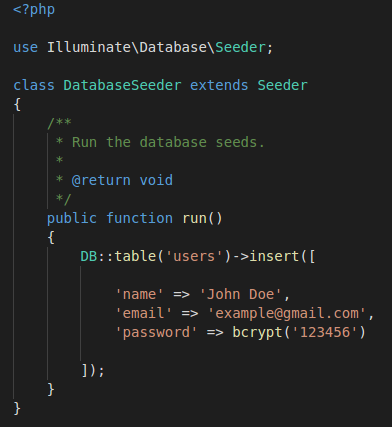
**2.3.2. Seed**

Dùng để tạo ra các dummy data(dữ liệu mẫu) cho database trong quá trình xây dựng ứng dụng. Sau khi đã đã có CSDL, việc tiếp theo bạn có thể cần làm đó là thêm một số dữ liệu mẫu trước khi bắt đầu thực hiện việc code các tính năng. Việc thêm dữ liệu mẫu chắc chắn sẽ được lặp lại nhiều lần, vì thế sẽ rất mất thời gian. Để giải quyết vấn đề này, [Seedẻr](https://laravel.com/docs/5.5/seeding#introduction) cung cấp các phương thức đơn giản để tạo ra dữ liệu mẫu bạn cần cho việc phát triển các tính năng và dễ dàng tái sử dụng.

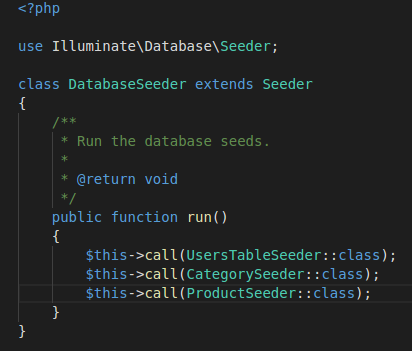
Việc tạo seeder tương tự như tạo migrate:



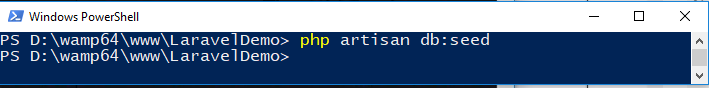
Sau khi tạo seeder, ta cần phải thêm dữ liệu vào.



Nếu bạn dẵ có những mẫu dữ liệu từ trước thì việc nhập liệu sẽ trở nên nhanh chóng vì chỉ cần gọi ra table nào cần thêm dữ liệu vào:



Cuối cùng là thực thi seeder bằng artisan



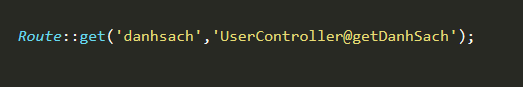
## **2.4. Route**

Nếu đã từng viết web bằng ngôn ngữ PHP bạn sẽ gặp tình trạng như sau: khi thực thi một tập tin php thì phần địa chỉ của trang web sẽ là ten\_file.php hoặc ten\_file.html, muốn đổi tên đường dẫn lại theo ý muốn của bàn thì bắt buộc phải sử dụng .htaccess để chuyển đổi tên đường dẫn theo ý muốn của mình. Điều này rất mất thời gian và quá trình sử dụng .htaccess không hề dễ chiu chút nào. Chính vì thế là Route ra đời.

Có thể tự tin nói rằng Route phần mạnh mẽ nhất của Laravel so với những framework khác.



Phía trên là các cách sử dụng route, các route có chung cấu trúc hai phần, một phần là đường dẫn mới, một phần là phương thức xử lý như sau: Route::Method(‘tên đường dẫn mới’,’xử lý’); . cách mà route hay được sử dùng nhất là sử dụng với Controller để gọi và thực thi các xử lý.

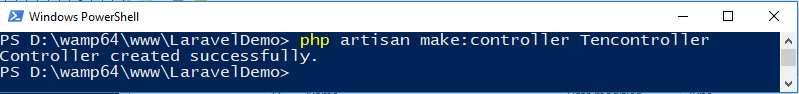


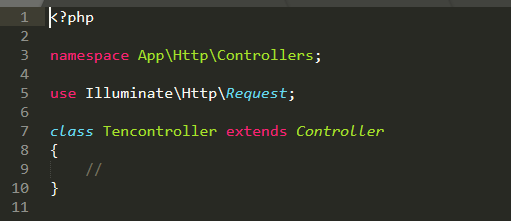
Khác với migrate hay seed, việc thực thi route không phải bằng artisan là là bằng trình duyệt web, cụ thể là trên thanh địa chỉ:



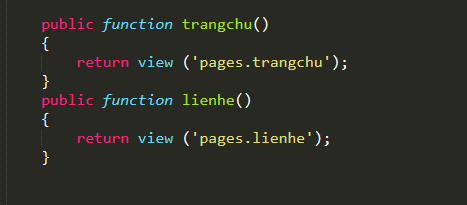
## **2.5. Controller**

Nếu như route phần giúp người dùng thấy được nội dung của trang web thì Controller chính là phần xử lý bên trong route để đưa ra được những nội dung nào sẽ hiện thị cho người dùng. Để dùng được controller, trước tiên ta cần phải tao ra nó bằng artisan:

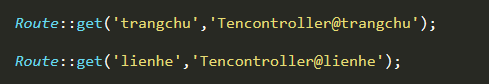




Sau khi có được file controller ta tiến hành gõ code xử lý. Ví dụ để hiện thi ra cho người dùng giao diện trang chủ hay trang liên hệ, ta làm như sau.



Để thực thi controller ta sử dụng nó khi tạo route:



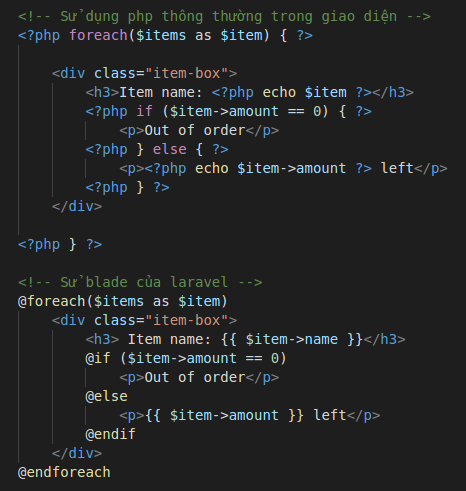
Ở ví dụ trên: khi ta truy cập vào địa chỉ *ten\_mien/trangchu* thì phần route có tên *trangchu* sẽ gọi đến hàm *trangchu()* trong file Tencontroller và hàm đó sẽ ra về giao diện của trang trang chủ hoặc liên hệ cho người dùng.

## **2.6. Blade template**

Template đó chính là 1 mẫu bố cục chung cho tất cả các trang có sử dụng lại những thành phần giống nhau mà không phải viết lại toàn bộ, từ đó trên mỗi trang, chỉ cần thay đổi ở một số nơi được chỉ định trên trang từ template.

Template có tác dụng giúp sạch đi những đoạn code PHP nằm trong View nên tách biệt hoàn toàn giữa người cắt CSS và người code PHP.

Blade rất đơn giản, nhưng lại đầy mạnh mẽ! Không giống những PHP template phổ biến khác, Blade không giới hạn chúng ta sử dụng code PHP trong views. Tất cả các file Blade sẽ được dịch thành file code PHP và cache cho đến khi file Blade bị thay đổi, điều đó cũng có nghĩa là Blade tự làm tất cả những việc cần thiết để có thể chạy views cho ứng dụng của bạn. Vd: trong file.php: <p><?php echo $email?></p> còn trong blade.php: <p>{{ $email }}</p>

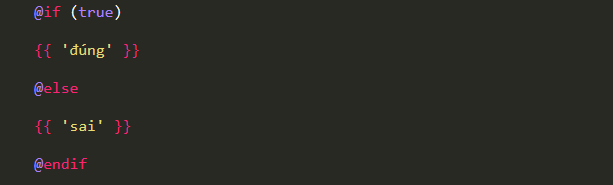


Việc tạo ra một file có dạng ten\_file.blade.php chỉ đơn giản là tạo mới một file ở bất kỳ dâu mà bạn thích với phần đuôi mở rộng là *.blade.php*

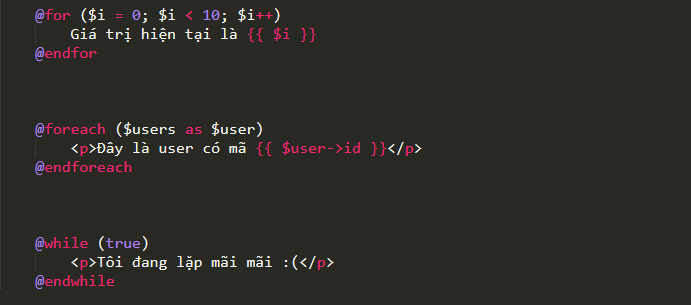
Một số cú pháp trong blade template thường dùng

+ Hiển thị biến: {{ $ten\_bien }}

+Lệnh if:

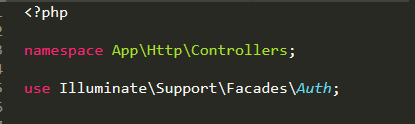


+ Vòng lặp:

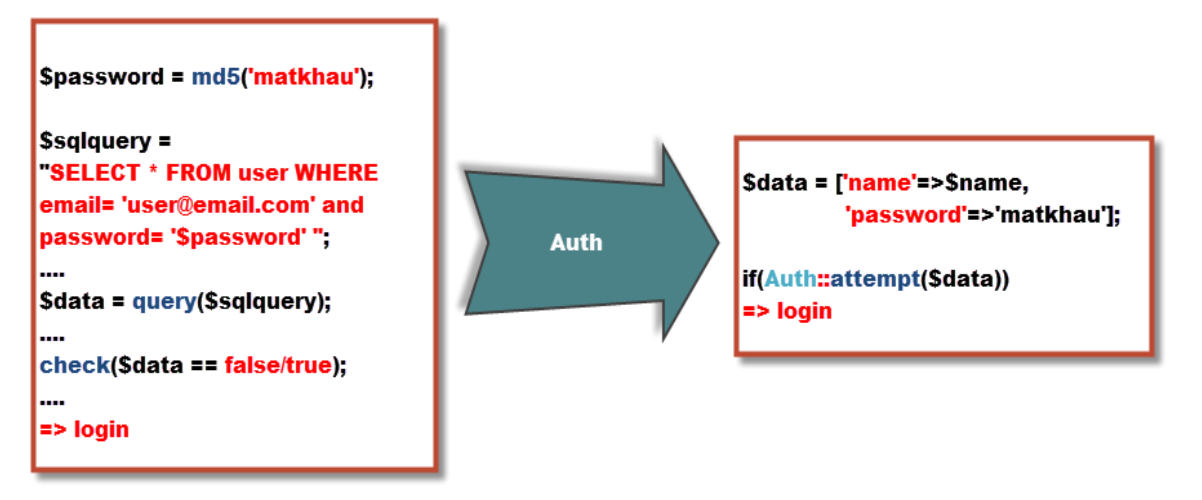


## **2.7. Authentication**

Authentication trong laravel giúp người dùng thực hiện và quản lý chức năng đăng nhập đăng xuất được thuận tiện và nhanh chóng. Để sử dụng được tính năng này bạn không cần phải tạo file hay cài đặt gì cả mà chỉ cần thêm thư viện vào file controller nà muốn xử lý đăng nhập, đăng xuất: use Illuminate\Support\Facades\Auth;



Nếu sử làm chức năng đăng nhập một cách thủ công sẽ tốn nhiều thời gian và dòng code xử lý như sau:



Trong Laravel bạn chỉ cần sử dụng Auth dã được khai báo thư viện là dễ dàng quản lý đang nhập.



## **2.8. Mô hình MVC**

Laravel được xây dựng và phát triển sẵn theo mô hình MVC (Model-View-Controller) nhờ đó mà cấu trúc và cách tổ chức code trong project được sắp xếp một cách hợp lý dễ dàng cho việc bảo trì cũng như phát triển về lâu dài.

Có thể hiểu một cách ngắn gọn nhất về MCV trong Laravel là: người dùng gửi yêu cầu đến Route bằng đường dẫn địa chỉ trên trình duyệt, Route nhận yêu cầu và truyền xuống cho Controller xử lý. Controller sẽ nhận yêu cầu xử lý và tiến hành lấy dữ liệu từ database nếu có sau đó trả dữ liệu lại cho View và View hiển thị lại nội dung cho người dùng.

