En esta tercera lección del Curso **Primeros Pasos con Laravel 5.\*** , comenzaremos con el estudio detallado de los principales componentes de Laravel. Este videotutorial lo dedicaremos a conocer la estructura de los directorios y archivos de un proyecto de Laravel y empezaremos a trabajar con las rutas.

Repositorio

Mira el código en GitHub: [**actual**](https://github.com/StydeNet/first-steps/tree/2), [**resultado**](https://github.com/StydeNet/first-steps/tree/3), [**comparación**](https://github.com/StydeNet/first-steps/compare/2...3).  
Si no tienes acceso al repositorio, solicítalo a: [**admin@styde.net**](https://styde.net/como-crear-rutas-en-laravel-5/)

Notas

* La estructura de los directorios de un proyecto de Laravel es la siguiente:
  + **app/** donde se encuentran las clases de modelos, controladores, rutas, entre otras.
  + **bootstrap/**contiene configuraciones del framework y cache.
  + **config/**donde se ubican los archivos de configuración del proyecto.
  + **database/**donde creamos las migraciones, seeders y model factories, es decir, lo relacionado a la base de datos.
  + **public/**contendrá todos los archivos que estarán disponibles para los usuarios.
  + **resources/**se encuentran los assets de nuestra aplicación
  + **storage/**archivos temporales del framework.
  + **tests/**para ubicar todas las pruebas que tendrá la aplicación.
  + **vendor/**donde se alojaran los componentes de terceros manejados por Composer.
* **mod\_rewrite** es un módulo del servidor web Apache que convierte las URL solicitadas en otra según una regla.
* **.htaccess** es un archivo de configuración que permite definir distintas reglas para directorios y archivos como por ejemplo limitar el acceso a directorios, crear URL más amigables, crear redireccionamientos, entre otras. En el caso de Laravel lo usa para manipular todas las peticiones que se hacen al proyecto y con la regla RewriteRule ^ index.php [L]  configura que las peticiones de tipo ruta sean enviadas a public/index.php para ser procesadas.
* Para crear una ruta en el proyecto vamos al archivo: **/app/Http/routes.php**
* La forma más simple de una ruta acepta los siguientes parámetros: la ruta (URI) y una función anónima (Clousure)

Route::get('/', function () {

return view('welcome');

});

Para conocer un poco más sobre [**Rutas con filtros en Laravel 5.1**](https://styde.net/rutas-con-filtros-en-laravel-5-1/)

Material Relacionado

* [**Flujo de trabajo en el desarrollo con Laravel**](https://styde.net/flujo-de-trabajo-en-el-desarrollo-de-laravel/)
* [**Instalación de Composer y Laravel 5.1**](https://styde.net/instalacion-de-composer-y-laravel-5-1/)
* Curso [**Administración de servidores para PHP y Laravel**](https://styde.net/curso-configuracion-administracion-de-servidores-php-laravel/)
* Serie [**Aprende Front-end (CSS, Sass)**](https://styde.net/aprende-front-end-css-sass/)