

Bienvenido a la segunda lección del Curso Primeros Pasos con Laravel 5.\*. En esta oportunidad, antes de empezar a aprender paso a paso cómo desarrollar aplicaciones con Laravel, te vamos a mostrar cómo es el flujo de trabajo para el desarrollo de un feature con Laravel de manera profesional, haciendo uso de las distintas herramientas y componentes que nos proporciona el framework.

En este punto ya deberías tener Laravel instalado, sino mira nuestro [videotutorial sobre instalación](https://styde.net/instalacion-de-composer-y-laravel-5-1/) o coméntanos tus dudas o posibles problemas sobre la instalación, dentro de dicho tutorial.

Importante: Este video es una especie de abreboca de lo que aprenderás dentro de éste y nuestros otros cursos. No te preocupes si no entiendes todo, pues a medida que avances en el curso, aprenderás más y con más calma sobre cada uno de estos temas. La idea es mostrarte el flujo de trabajo para desarrollar desde cero un feature de una aplicación de Laravel, que en esta lección será crear el listado de notas para una mini aplicación de notas.

**Notas**

* Laravel tiene un framework de pruebas integrado usando PHPUnit para hacer el desarrollo orientado a pruebas o TDD, del cual hemos hablado en:
  + [**Cómo escribir pruebas unitarias y de aplicación y por qué es importante**](https://styde.net/como-escribir-pruebas-unitarias-y-de-aplicacion-tipos-diferencias/)
  + [**Cómo funcionan las pruebas de integración en Laravel 5.1**](https://styde.net/como-funcionan-las-pruebas-de-integracion-en-laravel-5-1/)
  + [**Pruebas de integración en Laravel 5.1**](https://styde.net/pruebas-de-integracion-en-laravel-5-1/)
  + Además, el [**Curso introductorio de Laravel 5.1**](https://styde.net/curso-introductorio-laravel-5-1/) y el [**Curso: Crea tu primera aplicación con Laravel 5**](https://styde.net/curso-crea-aplicaciones-con-laravel-5/) tienes partes con lecciones dedicadas exclusivamente a pruebas en Laravel.
* Para conocer qué son y cómo se trabaja con los namespace en PHP: [**Qué es PSR-4 y uso de los namespaces**](https://styde.net/curso-de-laravel-5-que-es-psr-4-y-uso-de-los-namespaces/)**.**
* La configuración de la base de datos para un proyecto de Laravel se puede hacer creando un archivo llamado .env en el directorio raíz , que contenga las variables presentes en el archivo .env.example, asignando a las variables las credenciales para conexión de la base de datos que usará. Para un proyecto que se trabajará con [Homestead](https://styde.net/vagrant-homestead/) la configuración por defecto de la base de datos es la siguiente:

DB\_DATABASE=homestead

DB\_USERNAME=homestead

DB\_PASSWORD=secret

* En caso de ser otro [**ambiente de desarrollo**](https://styde.net/instalacion-y-configuracion-de-entornos-de-trabajo/) sustituye los datos por los correspondientes para tu proyecto.
* Las funciones helpers de Laravel son funciones de ayuda que cumplen tareas específicas, en el caso de dd() sirve para mostrar la variable dada y finalizar la ejecución del script. Para conocer más sobre helpers de laravel y cómo crear tus propios helpers: [**Cómo crear Helpers personalizados en Laravel**](https://styde.net/como-crear-helpers-personalizados-en-laravel/)
* Con la herramienta tinker, [**la consola interactiva de Laravel**](https://styde.net/tinker-consola-de-comandos-en-laravel/)podemos interactuar con la base de datos y los métodos de Laravel, por ejemplo, en el caso del video vaciar la tabla de la base de datos de un modelo. Primero se inicia ejecutando

**php artisan tinker**

y luego se interactua usando el método truncate() de esta manera:

**\App\Note::truncate();**

* Con la función de PHP [compact](http://php.net/manual/es/function.compact.php) se puede pasar a la vista un array asociativo producto de las variables que pasamos como argumentos.
* Larave además trae incluido un motor de plantillas para PHP llamado Blade con el cual podemos tener vistas con código más limpio, como por ejemplo:

**{{ $note->note }}**

**//reemplaza a**

**<?php echo $note->note ?>**

Bien, con este video pudiste ver cómo fue el desarrollo de un feature de una aplicación usando TDD: se creó la tabla de la base de datos, el modelo, su respectiva vista y la ruta en el proyecto y todo guiando por lo que nos indicaba la prueba hasta hacerla pasar y con eso garantizamos que está funcionando. ¿Te animas a aprender a desarrollar de esta manera? En los próximos videos continuarás conociendo paso a paso a trabajar con Laravel.

No olvides compartir en las redes sociales y si te interesa un tema en especifico lo puedes solicitar en [Teach Me](http://teach.styde.net/).

**Material relacionado**

* **Lección anterior:** [**Instalación de Composer y Laravel 5.1**](https://styde.net/instalacion-de-composer-y-laravel-5-1/)
* [**Artisan, la interfaz de línea de comandos de Laravel**](https://styde.net/artisan-interfaz-linea-comandos-de-laravel/)
* [**Alias de comandos para la consola Windows/Linux/Mac**](https://styde.net/alias-de-comandos-para-la-consola-windowslinuxmac/)