El objetivo de este material es mostrar el uso del Framework Larval en su versión 4.2, aunque para versiones posteriores puede que no hay una cambio tan notorio, primeramente Laravel es un Framework PHP, el cual facilita el desarrollo de aplicaciones web, ofreciendo grandes ventajas en comparación con la programación PHP puro.

Antes de comenzar es importante conocer el concepto de Framework, el cual consiste en una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de software concretos, que puede servir de base para la organización y desarrollo de software.

Un Framework en la mayoría de los caso pude incluir soporte de programas, bibliotecas y un lenguajes interpretado, entre otras herramientas que ayudan a desarrollar y unir diferentes componentes de un proyecto.

Entre las ventajas de la utilización del Framework Laravel de PHP está en la agilidad de desarrollo comparado, el uso del patrón MVC (Modelo Vista Controlador), lo cual permite tener una estructura de desarrollo más ordenada evitando mezcla un conjunto de tecnologías y como hojas de estilo, lenguaje del lado del cliente y servidor en un mismo archivo separando la lógica de las vistas, además de que facilita la extensibilidad o que el software siga creciendo sin tener impedimentos. También Laravel permite una mayor seguridad en nuestra aplicación ya que sus sentencias SQL se encuentran protegidas por defecto ante ataques, y es muy fácil utilizar protección contra CSRF en los formularios al igual que escapar caracteres inválidos.

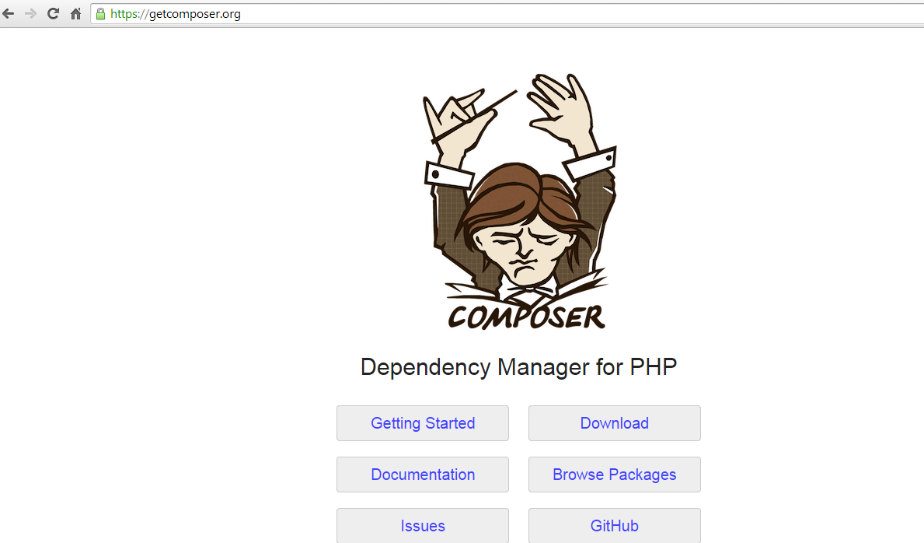
Otra característica muy interesante que cuenta Laravel es la utilización de modelos ORM los cuales tienen la finalidad de tratar las tablas de la base de datos como objetos lo que permite realizar las sentencias CRUD (CREATE READ UPDATE DELETE) de una manera muy sencilla y sin tener la necesidad de escribir las sentencias SQL, además otra ventaja es que el modelo ORM puede utilizar distintas bases de datos sin necesidad de modificar todo el código fuente algo sería imposible en una programación tradicional.

INSTALACIÓN

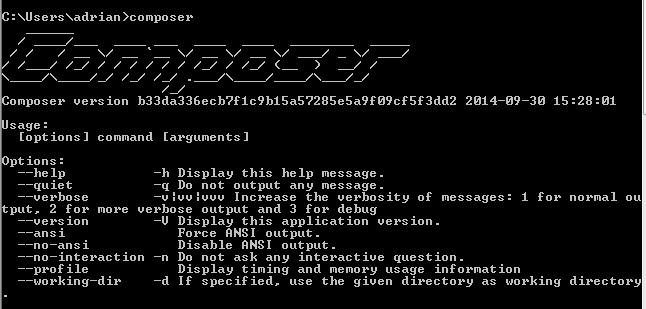
Para comenzar a utilizar el Framework Laravel de PHP, solamente debe de asegurarse que cuenta con los siguientes requerimientos:

* Tener una versión de PHP igual o superior a 5.3.
* Poseer el gestor de paquetes composer.

Composer es un gestor de paquetes de PHP que permite el manejo de dependencias PHP (librerías adicionales de PHP), similar al utilizado en otros Frameworks como PIP de Django, o para que se den una idea como el gestor de paquetes de distribuciones Linux donde escriben un comando en la terminal y este se encarga de localizar el paquete y descárgalo e instalarlo. Pueden descargarlo de la página http://www.getcomposer.org. Y los llevara a una página como esta.



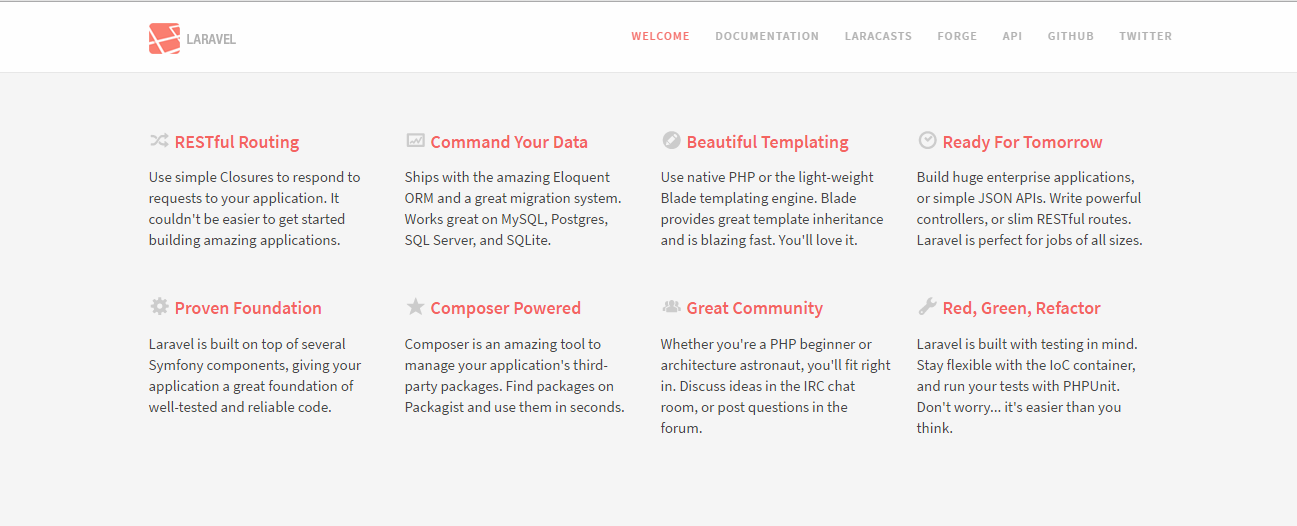
El proceso de instalación de composer en Windows es sumamente fácil solo es necesario definir la ubicación del archivo php.exe y dar un par de veces siguiente has que concluya la instalación. Una vez concluida la instalación es necesario comprobar que es posible utilizar el comando composer, para hacer esto habrá una terminal o consola y escriba la el comando composer si aparece un error deberá definrlo en variables del entorno, dirigiéndose a propiedades del sistema y configuración avanzada.



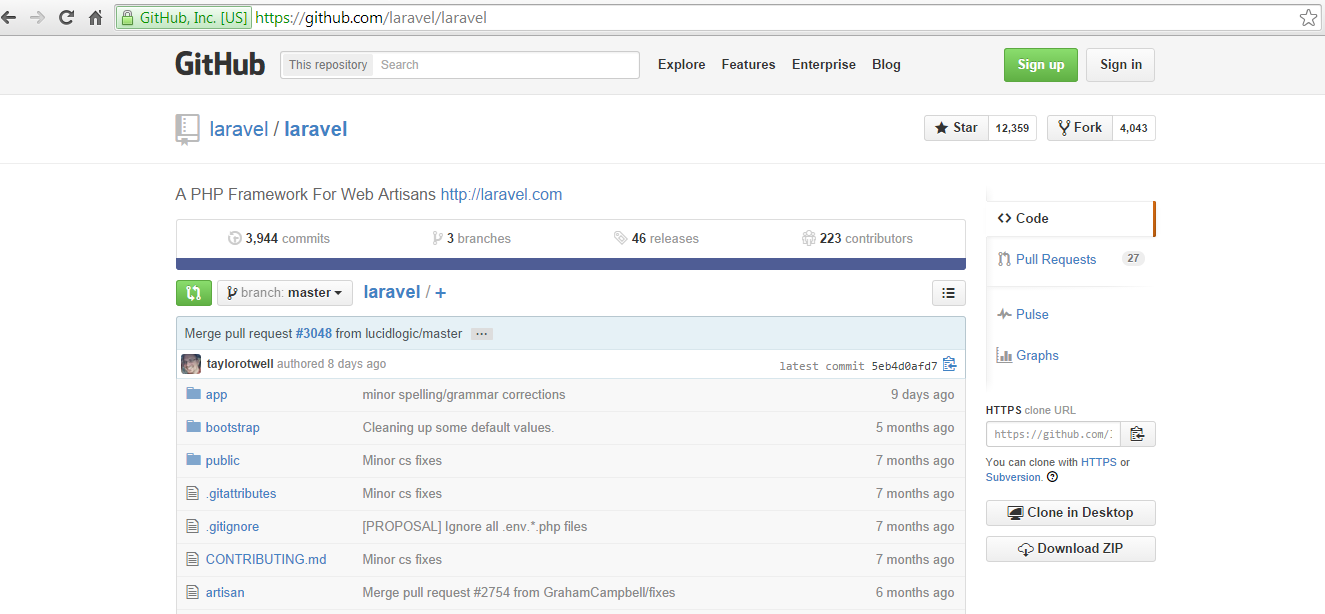
INSTALACIÓN DE LARAVEL

OPCIÓN 1

Para instalar Laravel existen dos maneras la primera es dirigiéndose a la página web de laravel en htpp://laravel.com y seleccionar Github. Esta acción lo dirigirá al repositorio público de Laravel el Github donde se encontrara la última versión de Laravel.



Una vez en Github solamente es necesario presionar el botón Download Zip y comenzarla la descarga de la última versión de Laravel.

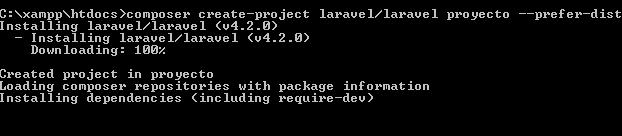


Una vez descargado en archivo Zip solamente falta descargarlo en la raíz de su sitio web de manera local el cual puede ser htdocs/ o www/ de acuerdo al paquete de servidor web que hayan instalado.

OPCIÓN 2

Para la instalación en esta opción usaremos el gestor de paquetes composer que instalamos anteriormente, el primer paso a realizar es abrir una consola y dirigirse a la carpeta htdocs/ o www/ de acuerdo a su máquina utilizando el comando cd y la carpeta.

Una vez en la carpeta es necesario ingresar el siguiente comando **composer create-project laravel/laravel nombre-del-proyecto –prefer-dist,** al igual que la instalación anterior descargara la última versión de composer en su sistema. Al presionar Enter comenzará la instalación.

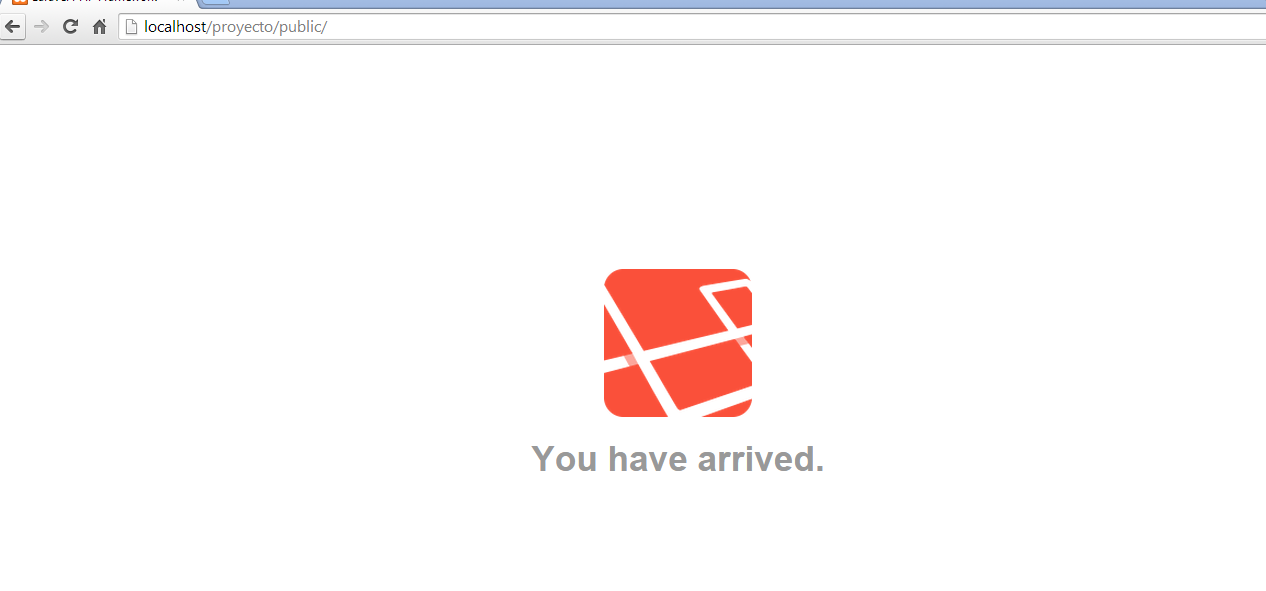


Una vez instalado Laravel por alguno de los pasos anteriores ya tienes instalado y utilizable el Framework Laravel.

ESTRUCTURA DE CARPETAS

COMPROBAR QUE LARAVEL ESTA FUNCIONANDO

Para comprobar que Laravel está corriendo en tu maquina solo basta con abrir algún navegador e ir a **localhost/nombre-de-tu-proyecto/public** y se mostrará una página como la siguiente, si aparece indica que todo está funcionado adecuadamente.

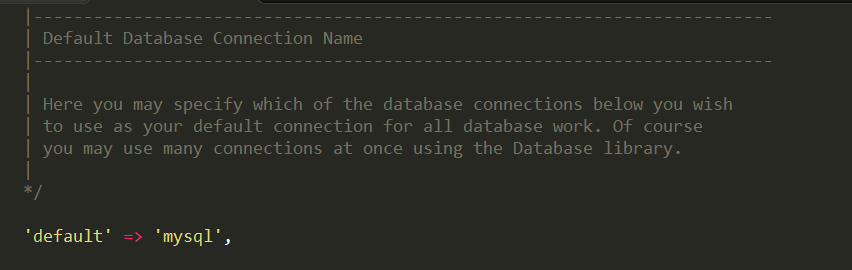


PRIMEROS PASOS CON LARAVEL

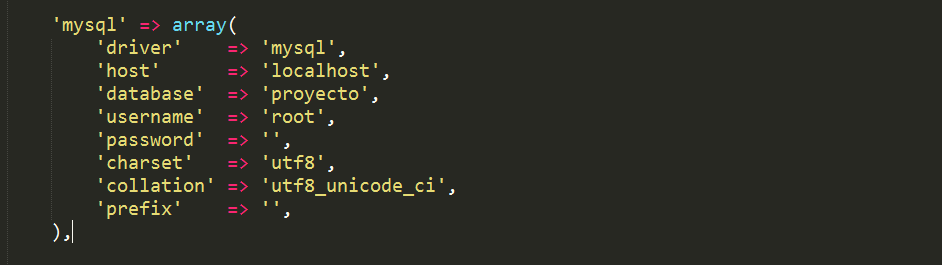
En este apartado se demostrará algunas de las funcionalidades más importantes que se pueden utilizar con el Framework Laravel, puede que parezca un poco complejo pero cuando profundice más en este documento lo encontrara más comprensible. En este ejemplo se planea tener una base de datos y obtener el primer registro obtenido, pero servirá para tener una mejor idea del proyecto y en que sección debería ir cada fragmento de código.

Para comenzar se necesita crear una base de datos, puede elegir entre varias opciones y en todas ellas el código seguirá siendo el mismo, en este caso usaremos mysql y la base de datos se llamara proyecto. Nada muy complejo.

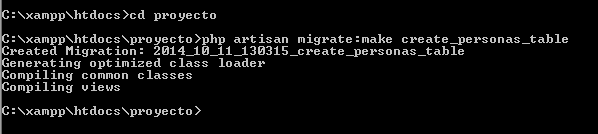
Ahora es necesario definir la base de datos a usar en nuestro proyecto para eso nos dirigimos a la carpeta **app/config/database.php**,por defecto esta seleccionado mysql como la base de datos a usar, si desea usar una distinta solo cámbielo por alguna de las demás opciones.



Además es necesario definir los parámetros de conexión a usar, entonces nos dirigimos un poco más abajo y encontraremos, las posibles conexiones a usar, y cambiamos la contraseña, usuario y base de datos de acuerdo a nuestra configuración, como se muestra. Y presionamos guardar.



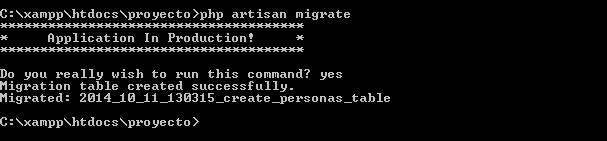
Una vez definida la base de datos, utilizaremos una herramienta que cuenta Laravel llamada migraciones lo que permite crear tablas en nuestra base datos y regresar de nuevo a su estado anterior, puede ser muy útil cuando no se tiene una base de datos bien definida o cuando el proyecto está siendo desarrollado por un grupo de personas siendo útil para tener un mejor control de los cambios. Para comenzar debemos utilizar nuevamente nuestra terminal y dirigirnos a la raíz de nuestro proyecto y teclear el siguiente comando **php artisan migrate:make create\_nombre\_de\_tu\_tabla\_create**, para tener un mejor control se recomienda utilizar un nombre en plural a la tabla a crear.



Ya que termino nos dirigimos a la siguiente estructura de carpetas **app/database/migrations/**  y dentro de ella encontraremos un archivo con la fecha actual vamos a abrirlo e ingresar el siguiente código.

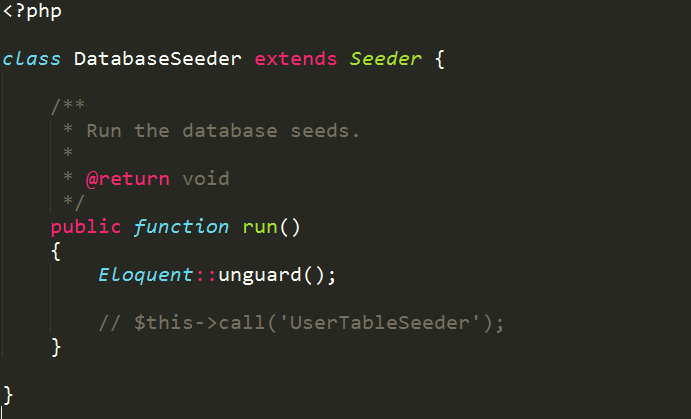


La función up es llamada cuando se ingresa el comando migrate y por consiguiente crear una tabla llamada personas, por defecto es necesario ingresar $table->timestamps() que Laravel utiliza aunque no se puede omitir, más adelante se describirá como. Y la función down será llamada cuando se quiera regresar a un estado anterior (borrar la tabla). Vamos a ver cómo usar esto. Volvemos a nuestra terminal e ingresamos el comando **php artisan migrate.**

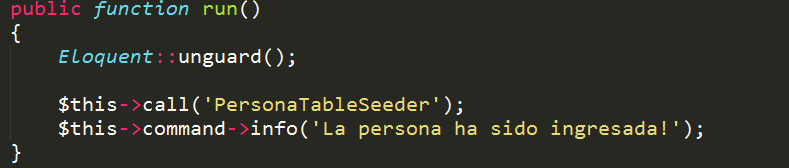
****

Ahora si comprueba podrá ver que ya cuenta con dos tablas la tabla que deseaba crear y la tabla migrations que se crear por defecto y es la encargada de volver las migraciones a un estado anterior o siguiente.

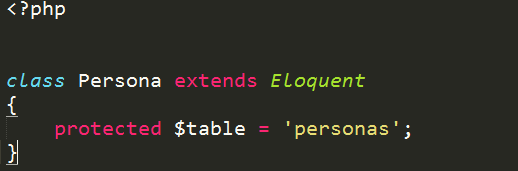
También es posible ingresar datos a nuestra tabla llamados seeds que es lo que realizaremos a continuación, para hacer esto es necesario ubicar el archivo **DatabaseSeeder.php** que se encuentra en **app/config/database/seeds/** este archivo tendrá un código como el siguiente:

cuenta

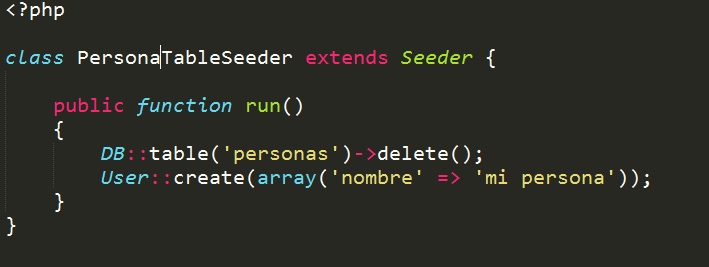
Como se muestra en la imagen anterior cuenta con una función llamada run, dentro de este método se debe ingresar la clase seed que se desea utilizar como se muestra en el código comentado en este caso escribiremos nuestra clase a utilizar y en la línea siguiente mostraremos un mensaje indicando que el ingreso de los datos fue exitoso. El código a usar quedaría de la siguiente manera.



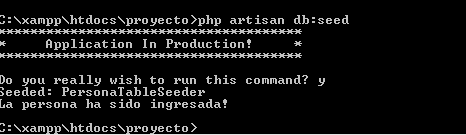
El siguiente paso será definir nuestro modelo el cual en una representación de nuestra clase (los modelos serán explicados detalladamente más adelante), para ello nos dirigiremos al siguiente directorio **app/models/** y crearemos el siguiente archivo **Persona.php** con el siguiente codigo.



Ahora solo bastara crear el archivo del seeder a utilizar cuyo nombre será el mismo que definimos en la función **call** en este caso **PersonaTableSeeder.php** que se ubicara en el directorio **app/config/database/seeds/**, en este archivo se ingresara el siguiente código.

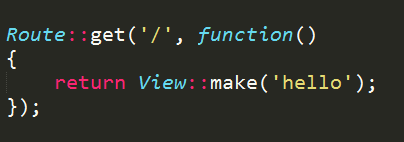


Una vez ingresado el código anterior volveremos a nuestra consola y escribiremos el siguiente comando **php artisan db:seed** .



Si todo ocurrió de la manera correcta debería de mostrarse al final el texto que describimos dentro del método run de la clase DatabaseSeeder, ahora puede comprobar en su base de datos y vera que ahora cuenta con una nueva tabla con un registro en ella, los concepto de migrations y seeder serán retomados más adelante.

Ahora que ya contamos con nuestra tabla en la base de datos podremos realizar consultas y mostrar registro en una página web, el primer paso para realizar cualquier tipo de acción en nuestro servidor tiene que existir una ruta que apunte a la acción que se realizara, para hacer esto es necesario definirlas en el archivo **routes.php** ubicado en la carpeta **app/** si lo abrimos solamente encontraremos la ruta **/**  que indica el home o la raíz similar al archivo index habitual que se ejecuta al ingresar a la página principal.



Como se muestra en la imagen es ruta cuenta con una función como parámetro y dentro de ella se encuentra la lógica de la aplicación en este caso simplemente regresa una vista que la pantalla de bienvenida de la aplicación.