**Identificación:** Anexo02-Laravel8  
**Versión:** 1.0

Tabla de contenido

[¿QUÉ ES LARAVEL? 2](#_Toc63095350)

[Modular y extensible 2](#_Toc63095351)

[Instalación Via Composer 2](#_Toc63095352)

[El instalador de Laravel 2](#_Toc63095353)

[Resumen del ciclo de vida 3](#_Toc63095354)

[HTTP / CONSOLA DEL KERNEL 3](#_Toc63095355)

[PROVEEDORES DE SERVICIOS 4](#_Toc63095356)

[RUTAS 4](#_Toc63095357)

[CONFIGURACIÓN 4](#_Toc63095358)

[Configuración del entorno 5](#_Toc63095359)

[**Configuración del Proyecto** 5](#_Toc63095360)

[Bibliografia 8](#_Toc63095361)

# **¿QUÉ ES LARAVEL?**

**Laravel** es un **framework**. En el 2011 su creador, Taylor Otwell, lanzo su primer versión. Por lo tanto, se trata de un framework con un enfoque fresco y moderno; ya que es bastante joven. Está hecho para arquitectura **MVC** y resuelve necesidades actuales como manejo de eventos y autenticación de usuarios. Cuenta con un código **modular** y **extensible** por medio de un administrador de paquetes y un soporte robusto para manejo de bases de datos.

## **Modular y extensible**

Laravel es modular y extensible. Esto quiere decir que te permite agregar todo lo que necesitas a través de su directorio Packalyst que cuenta con más de 5,500 paquetes. Esto con el objetivo de que siempre encuentres lo que necesitas.

## **Instalación Via Composer**

Si su computadora ya tiene PHP y Composer instalados, puede crear un nuevo proyecto de Laravel usando Composer directamente. Después de que se haya creado la aplicación, puede iniciar el servidor de desarrollo local de Laravel usando el comando serve de Artisan CLI:

*composer create-project laravel/laravel <proyecto> <versión>*

composer create-project laravel/laravel example-app

cd example-app

php artisan serve

## **El instalador de Laravel**

O puede instalar el instalador de Laravel como una dependencia global de Composer:

composer global require laravel/installer

laravel new example-app

php artisan serve

# **Resumen del ciclo de vida**

El punto de entrada para todas las solicitudes a una aplicación Laravel es el archivo public / index.php. Todas las solicitudes son dirigidas a este archivo por la configuración de su servidor web (Apache / Nginx). El archivo index.php no contiene mucho código. Más bien, es un punto de partida para cargar el resto del marco.

El archivo index.php carga la definición de autocargador generada por Composer y luego recupera una instancia de la aplicación Laravel desde bootstrap / app.php. La primera acción realizada por Laravel es crear una instancia del contenedor de la aplicación / servicio.

## **HTTP / CONSOLA DEL KERNEL**

A continuación, la solicitud entrante se envía al kernel HTTP o al kernel de la consola, según el tipo de solicitud que ingresa a la aplicación. Estos dos núcleos sirven como la ubicación central por donde fluyen todas las solicitudes. Por ahora, centrémonos en el kernel HTTP, que se encuentra en **app/Http/Kernel.php**.

El kernel HTTP extiende la clase **Illuminate\Foundation\Http\Kernel**, que define una matriz de bootstrappers que se ejecutarán antes de que se ejecute la solicitud. Estos bootstrappers configuran el manejo de errores, configuran el registro, detectan el entorno de la aplicación y realizan otras tareas que deben realizarse antes de que se maneje realmente la solicitud. Por lo general, estas clases manejan la configuración interna de Laravel de la que no debe preocuparse.

El kernel HTTP también define una lista de middleware HTTP por el que deben pasar todas las solicitudes antes de ser manejadas por la aplicación. Estos middleware manejan la lectura y escritura de la sesión HTTP, determinando si la aplicación está en modo de mantenimiento, verificando el token CSRF y más. Hablaremos más sobre esto pronto.

La firma del método para el método handle del kernel HTTP es bastante simple: recibe una solicitud y devuelve una respuesta. Piense en el kernel como una gran caja negra que representa toda su aplicación. Aliméntelo con solicitudes HTTP y devolverá respuestas HTTP.

## **PROVEEDORES DE SERVICIOS**

Una de las acciones de arranque del kernel más importantes es cargar los proveedores de servicios para su aplicación. Todos los proveedores de servicios para la aplicación están configurados en la matriz de proveedores del archivo de configuración config/app.php.

Laravel iterará a través de esta lista de proveedores y creará una instancia de cada uno de ellos. Después de crear una instancia de los proveedores, se llamará al método de registro en todos los proveedores. Luego, una vez que se hayan registrado todos los proveedores, se llamará al método de arranque en cada proveedor.

## **RUTAS**

Uno de los proveedores de servicios más importantes de su aplicación es App\Providers\RouteServiceProvider. Este proveedor de servicios carga los archivos de ruta contenidos en el directorio de rutas de su aplicación. Adelante, abra el código RouteServiceProvider y observe cómo funciona.

Una vez que se haya iniciado la aplicación y se hayan registrado todos los proveedores de servicios, la solicitud se entregará al enrutador para su envío. El enrutador enviará la solicitud a una ruta o controlador, además de ejecutar cualquier middleware específico de ruta.

El middleware proporciona un mecanismo conveniente para filtrar o examinar las solicitudes HTTP que ingresan a su aplicación. Por ejemplo, Laravel incluye un middleware que verifica si el usuario de su aplicación está autenticado. Si el usuario no está autenticado, el middleware redirigirá al usuario a la pantalla de inicio de sesión. Sin embargo, si el usuario está autenticado, el middleware permitirá que la solicitud continúe en la aplicación. Algunos middleware se asignan a todas las rutas dentro de la aplicación, como las definidas en la propiedad $ middleware de su kernel HTTP, mientras que algunos solo se asignan a rutas o grupos de rutas específicos.

# **CONFIGURACIÓN**

Todos los archivos de configuración para el marco de Laravel se almacenan en el directorio de configuración.

Estos archivos de configuración le permiten configurar cosas como la información de conexión de su base de datos, la información de su servidor de correo, así como otros valores de configuración básicos, como la zona horaria de su aplicación y la clave de cifrado.

## **Configuración del entorno**

A menudo es útil tener diferentes valores de configuración según el entorno donde se ejecuta la aplicación. Por ejemplo, es posible que desee utilizar localmente un controlador de caché diferente al que utiliza en su servidor de producción.

Para hacer esto fácil, Laravel utiliza la biblioteca PHP **DotEnv**. En una nueva instalación de Laravel, el directorio raíz de su aplicación contendrá un archivo **.env.example** que define muchas variables de entorno comunes. Durante el proceso de instalación de Laravel, este archivo se copiará automáticamente a .env.

El archivo **.env** predeterminado de Laravel contiene algunos valores de configuración comunes que pueden diferir según si su aplicación se está ejecutando localmente o en un servidor web de producción. Estos valores luego se recuperan de varios archivos de configuración de Laravel dentro del directorio de configuración usando la función env de Laravel.

Si está desarrollando con un equipo, es posible que desee continuar incluyendo un archivo .env.example con su aplicación. Al colocar valores de marcador de posición en el archivo de configuración de ejemplo, otros desarrolladores de su equipo pueden ver claramente qué variables de entorno son necesarias para ejecutar su aplicación.

### **Configuración del Proyecto**

Para configurar su proyecto deben ir a la ruta de su proyecto y buscar el archivo .env

* Nombre del Proyecto

APP\_NAME=”MI SENA”

* Entorno de Variable

APP\_ENV=development ó APP\_ENV=local

* Mostrar Errores

APP\_DEBUG=true

* URL del Proyecto

APP\_URL=http://localhost

* Motor de Base de Datos

DB\_CONNECTION=mysql

* Host de la Base de Datos

DB\_HOST=127.0.0.1

* Puerto de la Base de Datos

DB\_PORT=3306

* Nombre de la Base de Datos

DB\_DATABASE=hireme

* Usuario de la Base de Datos

DB\_USERNAME=root

* Clave de la Base de Datos

DB\_PASSWORD=wramirez

Texto

Descripción generada automáticamente

# Bibliografia

https://laravel.com/docs/8.x