

# Estudio Teorico Laravel

## Sumario

Objetivo.....	3
Introducción.....	4
Competidores.....	5
Características.....	6
Componentes, relación y utilidad.....	7
Webgrafía, fuentes, tutoriales, cursos, recomendados.....	8
Conclusión (teoría).....	9

## Objetivo

El objetivo de este estudio es aprender como funciona Laravel, su estructura y posibilidades que ofrece

## Introducción

Laravel es un framework de código abierto para el desarrollo de aplicaciones web basadas en PHP. Fue lanzado el 9 de junio de 2011.

Laravel cuenta con una estructura MVC (Modelo vista-controlador), su arquitectura es AMP

## Competidores

A pesar de que Laravel sea el framework de PHP mas usado, compite con varios frameworks, entre ellos:

CodeIgniter: Es facil de aprender y muy eficiente, por ello esta enfocado en el rendimiento

Symfony: Es usado en proyectos a gran escala, es mas complejo que Laravel pero proporciona una mayor flexibilidad y control. Laravel incorpora componentes de Symfony

Zend: Tiene la caracteristica de que permite cargar solo los componentes que vayan a usarse en la aplicacion, gracias a esto la reutilizacion de codigo se facilita mucho

Phalcon: Escrito en C, es un framework muy ligero y con un muy buen rendimiento, sin embargo, su documentacion no es tan completa como la de Laravel























CakePHP: Uno de los primeros frameworks MVC de PHP, cuenta con plantillas y validacion integrada, tiene una serie de configuraciones ya preestablecidas que agilizan la creacion de proyectos

## Características

Hablar de su estructura AMP, directorios que crea, etc

La última versión estable es Laravel 11, lanzada el 12 de Marzo de 2024

Nada mas crear un proyecto de laravel, se nos presentara esta estructura:

 app	14/03/2024 08:51	Carpeta de archivos	
 bootstrap	14/03/2024 08:51	Carpeta de archivos	
 config	14/03/2024 08:51	Carpeta de archivos	
 database	14/03/2024 10:32	Carpeta de archivos	
 public	14/03/2024 08:51	Carpeta de archivos	
 resources	14/03/2024 08:51	Carpeta de archivos	
 routes	14/03/2024 08:51	Carpeta de archivos	
 storage	14/03/2024 08:51	Carpeta de archivos	
 tests	14/03/2024 08:51	Carpeta de archivos	
 vendor	14/03/2024 10:32	Carpeta de archivos	
 .editorconfig	14/03/2024 08:51	Archivo de origen ...	1 KB
 .env	14/03/2024 10:32	Archivo ENV	2 KB
 .env.example	14/03/2024 08:51	Archivo EXAMPLE	2 KB
 .gitattributes	14/03/2024 08:51	Archivo de origen ...	1 KB
 .gitignore	14/03/2024 08:51	Archivo de origen ...	1 KB
 artisan	14/03/2024 08:51	Archivo	1 KB
 composer.json	14/03/2024 08:51	Archivo de origen ...	2 KB
 composer.lock	14/03/2024 10:31	Archivo LOCK	292 KB
 package.json	14/03/2024 08:51	Archivo de origen ...	1 KB
 phpunit.xml	14/03/2024 08:51	xmlfile	2 KB
 README.md	14/03/2024 08:51	Archivo de origen ...	5 KB
 vite.config.js	14/03/2024 08:51	JSFile	1 KB

Estos directorios y archivos corresponden a lo siguiente:

- `app`: Contiene la lógica principal de la aplicación
- `bootstrap`: Contiene los archivos de inicialización de la aplicación
- `config`: Contiene la configuración de la aplicación
- `database`: Contiene las migraciones de la base de datos
- `public`: Contiene el archivo `index.php` (índice), así como cualquier archivo público
- `resources`: Contiene los recursos de la aplicación
- `routes`: Contiene las rutas de la aplicación
- `storage`: Contiene los archivos de almacenamiento de la aplicación, como imágenes o audios
- `tests`: Contiene las pruebas de la aplicación
- `vendor`: Contiene las dependencias de la aplicación instaladas por Composer.
- `.env`: Archivo de entorno de la aplicación, que contiene la configuración específica de la máquina, como la información de la base de datos y la configuración de correo electrónico.
- `.env.example`: Ejemplo del archivo de entorno de la aplicación.
- `composer.json`: Archivo de configuración de Composer, que especifica las dependencias de la aplicación.
- `composer.lock`: Archivo generado por Composer que especifica las versiones exactas de las dependencias instaladas.
- `package.json`: Archivo de configuración de npm, que especifica las dependencias de JavaScript de la aplicación.
- `README.md`: Archivo de documentación de la aplicación.

## **Componentes, relación y utilidad**



## **Webgrafía, fuentes, tutoriales, cursos, recomendados**

Competidores de Laravel:

<https://www.hostinger.es/tutoriales/mejores-frameworks-php>

<https://colorvivo.com/mejores-frameworks-php-desarrollo-web/>

Características:

[https://www.tutorialspoint.com/laravel/laravel\\_application\\_structure.htm](https://www.tutorialspoint.com/laravel/laravel_application_structure.htm)

<https://codersfree.com/courses-status/aprende-laravel-avanzado/estructura-de-carpetas>

## **Conclusión (teoría)**