

Estructura del trabajo de los martes curso 25/26 (documentar siempre las fuentes)

FASE 1: ESTUDIO TEÓRICO (primer trimestre) .....	1
➤ Título.....	1
➤ Objetivo .....	1
➤ Introducción.....	1
➤ Competidores.....	1
➤ Características .....	1
➤ Componentes, relación y utilidad.....	2
➤ Webgrafía, fuentes, tutoriales, cursos, recomendados .....	2
FASE 2: CASO PRÁCTICO (segundo trimestre) .....	3
➤ Uso del repositorio para almacenar el trabajo realizado.....	3
➤ Implementación del trabajo en un entorno de producción .....	3
➤ Conclusión (práctica).....	3

## FASE 1: ESTUDIO TEÓRICO (primer trimestre)

LARAVEL 12 (última estable)

### ➤ Título

Estudio teórico sobre el framework Laravel usado para crear aplicaciones web en PHP

### ➤ Objetivo

Realizar un CRUD funcional de la tabla departamentos

### ➤ Introducción

Laravel es un framework de código abierto basado en PHP que se usa para **crear aplicaciones webs**.

### ➤ Competidores

En este apartado se mencionarán otros frameworks que también son usados para desarrollar código PHP y sus **principales características**.

- CodeIgniter: Framework que también usa la arquitectura MVC

- Framework ligero, pensado en el rendimiento
- Simple uso y buena documentación
- Symfony
  - Altamente flexible que permite configurar componentes individuales
  - Aprovecha la funcionalidad de prueba incorporada
  - Excelente documentación
- Zend
  - PHP orientado a objetos y arquitectura MVC
  - Permite el uso de solo los componentes que deseas ignorando todo lo demás
  - Fácil integración con bibliotecas externas
- Phalcon: Framework de PHP extraño, ya que su código fuente está escrito en C
  - Arquitectura MVC
  - Rendimiento fantástico y reducción de la sobrecarga de recursos gracias a C

## ➤ Características

Se basa en el patrón MVC. Cuenta con gran número de librerías además de una comunidad bastante amplia y activa en foros, muy útil si te encuentras con problemas. Está enfocado en el desarrollo del lado del servidor. La documentación oficial es excelente tanto para gente principiante como para usuario más expertos.

De la estructura Modelo-Vista-Controlador

- Modelo: Encargado de trabajar con los datos. Estos hacen uso de **Eloquent**. Eloquent es un ORM (mapeo de objetos relacional) que básicamente nos permite relacionar los datos de nuestra base de datos con entidades.
- Vista: Se encargan de la parte visual, es decir de mostrar el contenido. Laravel tiene su propio sistema de plantillas llamado blade.
- Controlador: Es la parte lógica de la aplicación, ejecutan los modelos y deciden que vistas se mostrarán

## ➤ Componentes, relación y utilidad

**Rutas:** Laravel usa dos ficheros encargados de manejar las rutas. Uno de ellos (web.php) registra las rutas de nuestro sitio web, mientras que el segundo (api.php) se enfoca en la creación de APIs.

En la definición de la ruta le podemos indicar con qué tipo de petición trabajará la ruta. Get para las peticiones estándar, o POST si queremos usar guardado de datos

### ARTISAN:

Artisan es una interfaz de líneas de comandos para trabajar en Laravel.

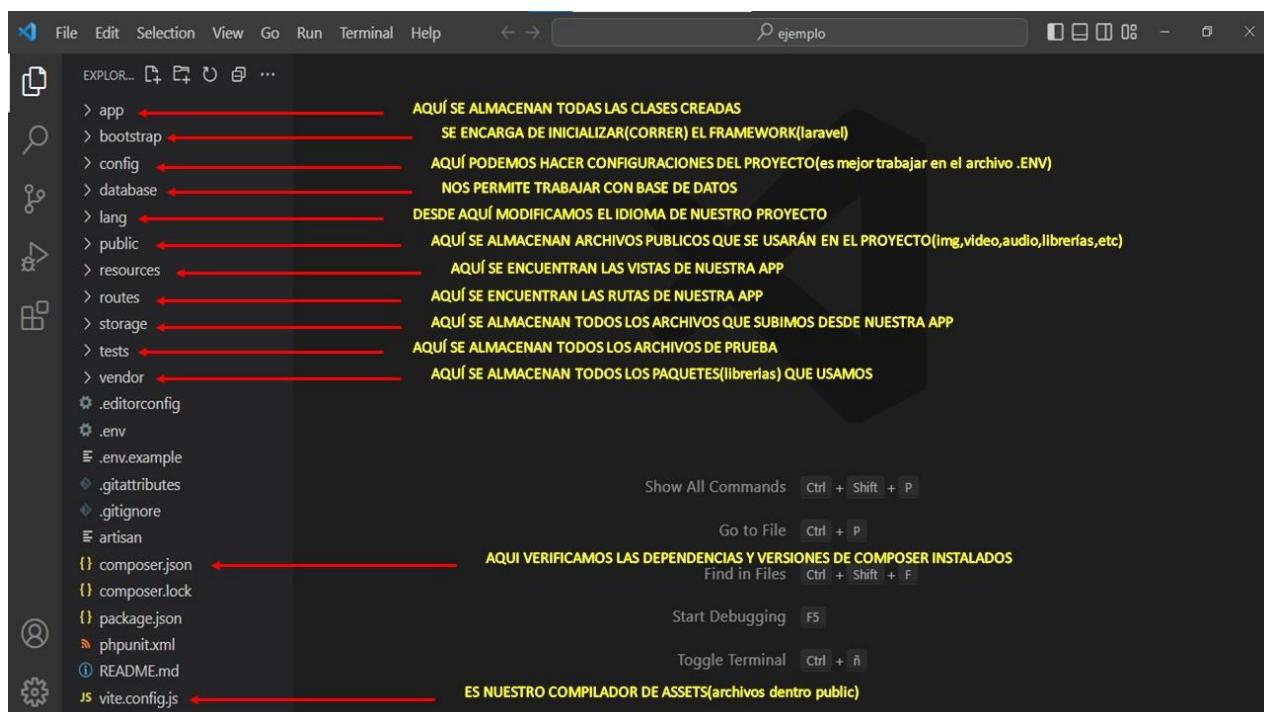
Además, Artisan también será el nombre del servidor que usa Laravel por defecto. Gracias a esto, no necesitaremos instalar un servidor web externo como Apache para poder ver nuestro trabajo.

## MIGRACIONES:

Las migraciones es una forma de gestionar los cambios en tu base de datos. Básicamente, son pequeños fragmentos de código que ejecutan acciones sobre la estructura de tu base de datos. Gracias a las migraciones puedes tener un control total sobre los cambios que se han ido realizando sobre la base de datos y revertir acciones.

## ESTRUCTURA DE DIRECTORIOS:

Laravel sigue la siguiente estructura de directorios:



En la imagen de arriba podemos ver los directorios que se encuentran en la raíz, pero dentro del directorio app tendremos que gestionar otra estructura.



Las carpetas en el directorio app se generan automáticamente según se vayan usando comandos Artisan, con la excepción de los directorios Http, Models y Providers que vienen por defecto, aunque permite su eliminación si se desea.

Enlace a la página oficial donde explica con detalle la estructura de directorios y cada directorio: <https://documentacionlaravel.com/docs/11.x/structure>

## ➤ Webgrafía, fuentes, tutoriales, cursos, recomendados

Página oficial: <https://laravel.com/>

Webgrafía:

<https://raiolanetworks.com/blog/laravel-que-es-y-como-funciona/#que-es-laravel>

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

Tutoriales:

- <https://laravel.com/docs/12.x/installation>
- <https://www.youtube.com/playlist?list=PLZ2ovOgdIkVtF2yQ2kiZetWWTmOQoUSG>
- <https://www.programador-web.com/2025/03/06/guia-laravel-12/>
- <https://laraveldaily.com/course/laravel-from-scratch>
- <https://medium.com/@techsolutionstuff/laravel-12-crud-tutorial-step-by-step-guide-c81b7c062029>

Tutoriales que son de versiones diferentes:

- <https://codersfree.com/cursos/aprende-laravel-desde-cero>
- <https://bitfumes.com/courses/laravel/laravel-beginner-tutorials>

## ➤ Conclusión (teoría)

# FASE 2: CASO PRÁCTICO (segundo trimestre)

➤ Preparación del entorno de desarrollo

➤ Desarrollo de una ( aplicación web (Ej: Mantenimiento de una tabla con control de

- Uso del repositorio para almacenar el trabajo realizado
- Implementación del trabajo en un entorno de producción
- Conclusión (práctica)

#### RESUMEN

- Presentación resumen