

# **Arquitecturas web y frameworks**

## 1. Arquitectura WEB

La arquitectura de desarrollo web son los cimientos sobre los se va a ir desarrollando nuestro sitio web final, es el conjunto de lenguajes y tecnologías que va a utilizar el sitio web en su totalidad, siendo las más comunes LAMP y MEAN.

### 1.1. LAMP

La arquitectura LAMP se refiere a aquella arquitectura web más clásica que se compone de

- Linux, como sistema operativo.
- Apache, como servidor WEB.
- MySQL/MariaDB, como gestor de bases de datos.
- PHP/Perl/Python, como lenguaje de programación en servidor.

### 1.2. MEAN

La arquitectura MEAN o Stack Mean tiene como factor principal el uso de Javascript en todas las capas de desarrollo, tanto en cliente como en servidor.

Las tecnologías y lenguajes utilizados son los siguientes:

- MongoDB, es un sistema de base de datos NoSQL, que almacena los datos en estructuras, los cuales están definidos con la notación JSON.
- Express, es un módulo de NodeJS y como tal funciona sobre esta plataforma; este módulo ofrece los métodos suficientes en JavaScript, para poder manejar las solicitudes o peticiones que se hacen por medio de los métodos del protocolo HTTP.
- Angular, es un framework que facilita la manipulación del DOM, y por lo tanto en el mean stack es la plataforma que se usa para trabajar en el front end.
- NodeJS, Es la plataforma encargada del funcionamiento del servidor, que funciona totalmente con JavaScript.

### 1.3. Comparativa LAMP y MEAN

La diferencia de estas dos arquitecturas se encuentra en las tecnologías y lenguajes que utilizan.

En cuanto al Sistema Operativo, si bien LAMP solo funcionaría correctamente en Linux, MEAN puede ser instalado en cualquiera, sin poner ningún tipo de restricción.

Atendiendo al Servidor, LAMP utiliza una tecnología muy probada y madura como Apache, por otro lado, MEAN cumple esta funcionalidad mediante NodeJS, que, además de ser una tecnología muy joven, por lo que quien lo vaya a utilizar deberá conocer bien el entorno donde se va a ejecutar, siendo para esto un requisito importante tener un conocimiento medio de JavaScript. Además, dado que NodeJS restringe todo al uso de JavaScript, puede crear complejidades si hay que adaptar un entorno ya creado a esta nueva arquitectura.

En relación con el código, MEAN hace las cosas más sencillas al desarrollador, ya que, mediante Express y Angular gestiona todo el código mediante Javascript, tanto en cliente,

como en servidor, en lugar de necesitar conocer, además, PHP o Python por ejemplo en una arquitectura LAMP.

Por último, en cuanto al gestor de bases de datos, MEAN está diseñado para trabajar con bases de datos no relacionales, aunque existen plugins para facilitar la integración de bases de datos relacionales. Mientras que LAMP solo admite bases de datos relacionales.

En resumen, Trabajar con MEAN o pasarnos a este stack de aplicaciones nos traerá numerosos beneficios, de los que destacaremos el uso de un único lenguaje de programación como es JavaScript para absolutamente todo, flexibilidad en el despliegue de aplicaciones y mejorar la velocidad con la que se requieren los datos gracias a MongoDB. Por otro lado, si deseamos crear un sitio nuevo con arquitectura MEAN en un entorno ya creado, tendríamos que modificar los sitios ya existentes a la nueva arquitectura, ya que solo admite Javascript.

## 2. Framework

### 2.1. Descripción de framework

Un Framework, que se podría traducir aproximadamente como marco de trabajo, es el esquema o estructura que se establece y que se aprovecha para desarrollar y organizar un software determinado.

Sirve para poder escribir código o desarrollar una aplicación de manera más sencilla. Es algo que permite una mejor organización y control de todo el código elaborado, así como una posible reutilización en el futuro.

### 2.2. Laravel

Laravel tiene como objetivo ser un framework que permita el uso de una sintaxis elegante y expresiva para crear código de forma sencilla y permitiendo multitud de funcionalidades. Intenta aprovechar lo mejor de otros frameworks y aprovechar las características de las últimas versiones de PHP.

Gran parte de Laravel está formado por dependencias, especialmente de Symfony, esto implica que el desarrollo de Laravel dependa también del desarrollo de sus dependencias.

Sus principales características son las siguientes:

- Sistema de ruteo, también RESTful, que permite utilizar métodos HTTP de forma explícita, exponer las URIS como estructuras de directorios y transferir XML, JSON o ambos.
- Blade, Motor de plantillas
- Peticiones Fluent, un constructor de consultas SQL y está encargado de generar cualquier consulta a la base de datos
- Eloquent ORM( mapeo objeto-relacional), es una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y la utilización de una base de datos relacional como motor de persistencia
- Basado en Composer, que es un sistema de gestión de paquetes para programar en PHP.

- Soporte para el caché
- Soporte para Modelo Vista Controlador (MVC), que es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.
- Usa componentes de Symfony
- Adopta las especificaciones PSR-2 y PSR-4 , que son estándares de programación orientados a facilitar la lectura de éste entre distintos desarrolladores.

### 2.3. Motivo de elección de Laravel

El motivo de mi elección de Laravel como framework se debe a que, al ser uno de los frameworks más requeridos por las empresas y, dado que facilita enormemente el trabajo al desarrollador, y no es especialmente complicado de aprender, podríamos decir que tiene mucho futuro.