

# ARQUITECTURA WEB Y FRAMEWORK



**Autora:** Salomé Roldan

**Curso:** 2º Desarrollo de Aplicaciones Web

**Asignatura:** Desarrollo Web en Entorno Servidor

**Profesor:** Manuel Alejandro Romero Miguel

## Arquitecturas web

Las arquitecturas en el desarrollo web son proporcionadas por un servidor Web y utilizada por usuarios que se conectan desde cualquier punto vía clientes Web (browsers o navegadores). La arquitectura de un Sitio Web tiene tres componentes principales:

- Un servidor Web
- Una conexión de red
- Uno o más clientes

El servidor Web distribuye páginas de información formateada a los clientes que las solicitan. Los requerimientos son hechos a través de una conexión de red, y para ello se usa el protocolo HTTP. Una vez que se solicita esta petición mediante el protocolo HTTP y la recibe el servidor Web, éste localiza la página Web en su sistema de archivos y la envía de vuelta al navegador que la solicitó.

Las aplicaciones Web están basadas en el modelo Cliente/Servidor que gestionan servidores web, y que utilizan como interfaz páginas web.

La colección de páginas son en una buena parte dinámicas (ASP, PHP, etc.), y están agrupadas lógicamente para dar un servicio al usuario. El acceso a las páginas está agrupado también en el tiempo (sesión).

Los componentes de una aplicación Web son: Lógica de negocio, administración de los datos, interfaz.

Las aplicaciones web se modelan mediante lo que se conoce como modelo de capas, Una capa representa un elemento que procesa o trata información.

Los tipos son:

Modelo de dos capas: La información atraviesa dos capas entre la interfaz y la administración de los datos.

Modelo de tres Capas: Está diseñada para superar las limitaciones de las arquitecturas ajustadas al modelo de dos capas, introduce una capa intermedia (la capa de proceso).

Las capas de este modelo son: Capa de presentación (parte en el cliente y parte en el servidor), capa de proceso (servidor web), capa de datos (servidor de datos).

**La arquitectura LAMP**, promocionada por la editorial O'Reilly hasta desembocar en la adopción del término mencionado para referirse a ella, puede que sea uno de los casos de éxito más sobresalientes del software libre como arquitectura.

Esta arquitectura, que consta de un sistema operativo, un servidor web, una base de datos y un lenguaje de programación, todos ellos distribuidos bajo una licencia de software libre, ofrece una manera tremendamente rápida para la construcción de una aplicación web para usuarios finales a un bajo coste. Siendo tecnologías sin dependencia alguna entre ellas (nunca fueron pensadas con el fin de interoperar entre sí), hace que resulte sorprendente los buenos resultados que se suelen obtener en producción, y siendo un gran ejemplo de la variedad y calidad de productos que ofrece el software libre. Pero sus detractores también ofrecen

argumentos para no utilizarlo como por ejemplo la posible falta de soporte respecto a un Windows Server y una base de datos Oracle.

Por lo dicho anteriormente, las ventajas de LAMP son:

- Es una tecnología madura
- Está bien documentada
- Funciona bien en apps web de tamaño medio y grande donde la gestión de los datos en tiempo real no es vital
- Tiene una comunidad muy grande
- PHP es el lenguaje de servidor más utilizado del mundo, lo usan WordPress, Drupal, Prestashop, Magento, Moodle ...

Aunque tiene algunos inconvenientes:

- No funciona bien en aplicaciones de tiempo real que manejan grandes cantidades de datos de forma concurrente
- No está de moda
- Es más lento que MEAN
- Emplea varios lenguajes de programación

Una alternativa sería MEAN, ya que tiene varias ventajas:

- Lo utilizan las startups más punteras
- Es la mejor solución para programar aplicaciones de tiempo real que manejan grandes cantidades de datos de forma concurrente
- Es más rápido que LAMP
- Emplea un solo lenguaje para todo: JavaScript
- Está de moda

Aunque como todas las arquitecturas tiene desventajas que son:

- La orientación a eventos no es siempre el mejor marco para desarrollar aplicaciones
- MEAN está todavía en fase de investigación y desarrollo, lo que lo convierte en un entorno extremadamente volátil
- Cada dos por tres salen nuevas librerías que no siempre funcionan bien
- La documentación no es tan buena como en LAMP
- Muchas empresas pequeñas y medianas no pueden permitirse el riesgo de utilizar tecnologías tan cambiantes

Para aprender programación web puedes elegir tanto LAMP como MEAN. Sería recomendable combinar cosas de los dos mundos porque al final suele emplearse una u otra pila dependiendo de la aplicación a implementar.

Aunque **LAMP** es recomendable si no queréis arriesgar demasiado dinero invirtiendo en recursos, formación, personal, etc., en un entorno tan extremadamente cambiante.

**MEAN** es un acrónimo para denominar a la pila de tecnologías que, funcionando de forma conjunta, conforman un marco de trabajo full stack para desarrollo web, empleando JavaScript o alguna de sus variantes como lenguaje de programación.

Las tecnologías que forman la pila MEAN son las siguientes:

- MongoDB: Es un sistema gestor de bases de datos que se caracteriza por ser de tipo no relacional, porque almacena datos empleando documentos al estilo JSON junto a esquemas, en lugar de tablas. Los datos se almacenan en colecciones, los documentos de una misma colección no tienen por qué seguir exactamente el mismo esquema. Algunas de las características de Mongo son: Indexación, consultas ad-hoc, balanceo de carga, agregación y ejecución de JavaScript del lado del servidor.
- Express: Es un framework diseñado para la construcción de APIs y aplicaciones web, tiene como fundamentos ser mínimo y flexible, a la vez que provee de un conjunto robusto de funcionalidades para facilitar el desarrollo de aplicaciones. Entre dichas funcionalidades se encuentran métodos y utilidades HTTP que facilitan la creación de APIs REST.  
Algunas de las funcionalidades que provee Express son las siguientes:  
Direccionamiento, funciones de middleware, motores de plantilla.
- Angular: Es un framework JavaScript de código abierto, pensado para la creación de aplicaciones de una sola página basadas en el navegador y con capacidad MVC. Angular se encuentra actualmente en desarrollo y bajo constante evolución; aunque mantiene algunos conceptos de AngularJS, es incompatible con dicha versión, ya que no comparten principios ni estructura. La característica más importante de Angular 2 es su arquitectura basada en componentes, que trata de facilitar los principios básicos de bajo acoplamiento y alta cohesión.  
Esta aproximación aporta ventajas en cuanto a: Reutilización, facilidad para realizar pruebas, legibilidad, mantenibilidad.
- Node.js: es un entorno de ejecución multiplataforma, de código abierto, basado en ECMAScript y en el motor, que se caracteriza por ser asíncrono, y presentar un modelo de E/S orientado a eventos, no bloqueante. Resulta muy útil para la creación de programas altamente escalables como son los servidores web. Presenta un único hilo de ejecución, de forma que el orden de evaluación es secuencial, sin embargo, las operaciones relacionadas con entrada o salida de datos, se ejecutan de forma totalmente concurrente, gracias al obligado uso de callbacks. En cuanto al motor V8, se trata de un entorno de ejecución para JavaScript, desarrollado por Google para utilizarlo en Google Chrome; es software libre y se caracteriza porque compila el código fuente JavaScript a código máquina, en lugar de interpretarlo.  
Node.js incorpora una serie de módulos básicos, como pueden ser los relacionados con tareas de red, ficheros, buffers, temporizadores o streams.

La tecnología JSP es un acrónimo de Java Server Pages, que en castellano vendría a decir algo como Páginas de Servidor Java. Es una tecnología orientada a crear páginas web con programación en Java.

Con JSP podemos crear aplicaciones web que se ejecuten en variados servidores web, de múltiples plataformas, ya que Java es en esencia un lenguaje multiplataforma. Las páginas JSP están compuestas de código HTML/XML mezclado con etiquetas especiales para programar

scripts de servidor en sintaxis Java. Por tanto, las JSP podremos escribirlas con nuestro editor HTML/XML habitual.

## Framework

Symfony es un framework PHP de tipo full-stack construido con varios componentes independientes creados por el proyecto Symfony.

Sus principales características son:

- Su código, y el de todos los componentes y librerías que incluye, se publican bajo la licencia MIT de software libre.
- La documentación del proyecto también es libre e incluye varios libros y decenas de tutoriales específicos.
- Se puede adaptar para la mayoría de proyectos, aunque está pensado para grandes aplicaciones web, también se puede usar para otras más pequeñas.
- Aprender a programar con Symfony te permite acceder a una gran variedad de proyectos: el framework Symfony2 para crear aplicaciones complejas, el micro framework Silex para sitios web sencillos y los componentes Symfony para otras aplicaciones PHP.
- Symfony es el proyecto PHP más activo, lo que garantiza que nunca te quedarás atrapado en un proyecto sin actividad.
- Los componentes de Symfony son tan útiles y están tan probados, que proyectos tan gigantescos como Drupal 8 están contruidos con ellos.
- En todo el mundo se celebran varias conferencias dedicadas exclusivamente a Symfony. Para que te hagas una idea del tamaño de la comunidad, la conferencia Symfony española es el evento PHP más grande del país.

Tiene diferentes ventajas como:

- Es extremadamente flexible: se adapta a casi cualquier necesidad, permitiendo instalar únicamente las piezas requeridas para el proyecto en vez de todo el framework.
- Es tremendamente ampliable: según GitHub, es el proyecto php más activo, lo que te garantiza encontrar paquetes para prácticamente cualquier funcionalidad.
- Es un sistema estable: Garantiza que cada versión de Symfony recibirá soporte (actualizaciones y solución de problemas) durante tres años, además de compatibilidad con las versiones secundarias.
- Un sistema rápido y que consume poca memoria: Symfony ha sido desarrollado con la idea de ofrecer aplicaciones de alto rendimiento, es más rápido que otros frameworks usando la mitad de la memoria.
- Facilidad de uso: Gracias a la gran cantidad de documentación y tutoriales que se pueden encontrar en la web, cualquier profesional o usuario avanzado puede aprender rápidamente los conceptos más básicos de Symfony.

Symfony es una gran herramienta que conocía anteriormente para proyectos en PHP sin importar el alcance de los mismos aunque si quieres aprender PHP con esta herramienta es muy difícil. Considero que ahora ha logrado la estabilidad, ya que grandes desarrolladores van trabajando para facilitar más la cantidad de código que se necesita escribir y logrando quizá lo más importante que siempre se buscó.