

# DWES01\_TAREA02

## Investiga sobre arquitecturas web

### Descripción de arquitecturas en el desarrollo web

Describe además de la arquitectura **LAMP** alternativas a ella por ejemplo **MEAN**

Habla de las tecnologías y lenguajes de programación en **MEAN**

Realiza una comparativa con **LAMP (WAMP)**

Habla si ves alguna otra arquitectura o tecnología que te resulte interesante por ejemplo la que emplea **JSP**

- LAMP => **L**inux **A**pache **M**ysql **P**HP
- MEAN => **M**ongo **E**xpress **A**ngular **N**ode js

### LAMP



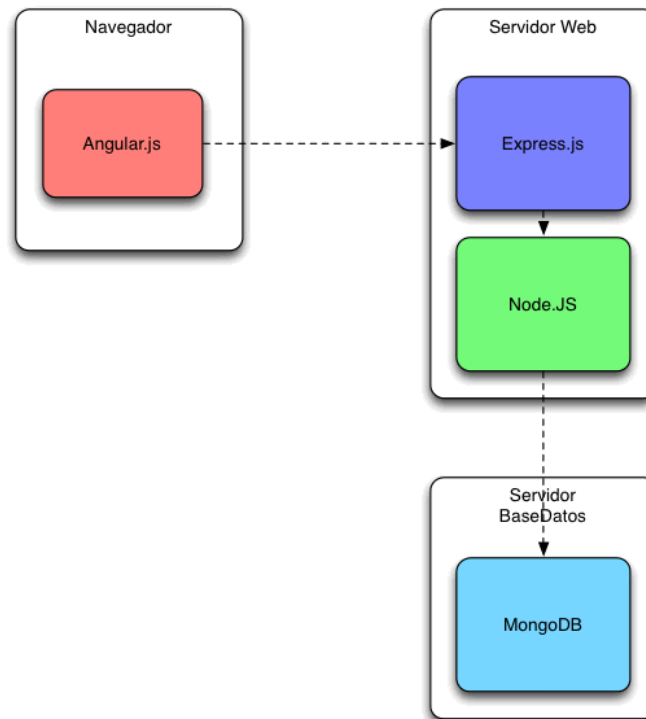
- **Linux**: Sistema operativo libre y gratuito que es utilizado por la gran mayoría de servidores conectados a Internet.
- **Apache**: Servidor HTTP uqe nos permite su utilización en entornos básicos o en otros más complejos ampliando funcionalidades con los diferentes módulos existentes.  
**MySQL / MariaDB**: Gestores de bases de datos.
- **PHP, Perl, Python**: Lenguajes de programación para el funcionamiento de las aplicaciones.

Las funcionalidades y rendimiento de la estructura LAMP, unida a su calidad de software libre y gratuito han motivado su gran difusión así como el origen de variantes.

### MEAN (ACRÓNIMO DE MongoDB, Express.js, Angular.js y Node.js)

- **MongoDB**: base de datos NoSQL.
- **Express.js**: framework JavaScript Web del Servidor.

- **Angular JS:** framework JavaScript del Cliente.
- **Node.js:** Permite ejecutar JavaScript del lado del servidor (servidor de aplicaciones).



Elementos fuertemente ligados al mundo JavaScript.

## TECNOLOGÍA Y LENGUAJES DE PROGRAMACIÓN EN MEAN

**MongoDB:** es un sistema de base de datos NoSQL, que almacena los datos en estructuras o “documentos”, los cuales están definidos con la notación JSON (Notación simple de objeto tipo JavaScript), lo que permite una rápida manipulación y transferencia de los datos. La mayor característica de esta plataforma es su escalabilidad, lo que significa que puede aumentar en forma considerable la cantidad de datos que almacena sin que esto afecte su funcionamiento en general.

**ExpressJS:** 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 (GET, POST, etc.). También ofrece un sistema simple de enrutamiento, que dentro del mean stack es aprovechado en el back-end o en el lado del servidor:

**AngularJS:** es un framework que facilita la manipulación del DOM ('Modelo de Objetos del Documento' o 'Modelo en Objetos para la Representación de Documentos), y por lo tanto en el mean stack es la plataforma que se usa para trabajar en el front end. Este framework permite crear una gran variedad de efectos, de una forma sencilla, reduciendo contundentemente la cantidad de código, lo que permite que sea mucho más sencillo de mantener.

**NodeJS:** Es la plataforma encargada del funcionamiento del servidor. Funciona totalmente con JavaScript, un lenguaje de programación que en un principio era dedicado a correr en el lado

del cliente, pero su uso se ha ampliado considerablemente en todos los aspectos de un sitio web.

## COMPARATIVA LAMP WAMP

Las dos estructuras comparten tres de los componentes: Apache, MySQL y Php. La diferencia radica en el tipo de sistema operativo sobre el que se instalan: Linux o Windows. En este sentido parece que las diferencias que podemos aportar sobre estas arquitecturas estarían más asociadas a las diferencias entre esos dos sistemas operativos. Junto a ellos podríamos mencionar MAMP, sistema análogo que corre en Macintosh.

## TECNOLOGÍA UTILIZADA POR JSP

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

La tecnología JavaServer Pages permite generar contenido Web dinámico como, por ejemplo, archivos HTML, DHTML, XHTML y XML, para incluirlos en una aplicación Web. Los archivos JSP son una forma de implementar contenido de páginas dinámico del lado del servidor. Los archivos JSP permiten a un servidor Web como, por ejemplo, Apache Tomcat, añadir contenido dinámicamente a las páginas HTML antes de enviarlas al navegador que las solicita.

Cuando se despliega un archivo JSP en un servidor Web que proporciona un motor de servlets, se procesa previamente en un servlet que se ejecuta en el servidor. Esto contrasta con JavaScript™ en el lado del cliente (dentro de códigos <SCRIPT>), que se ejecuta en un navegador. Una página JSP resulta ideal para tareas cuya ejecución es más adecuada en el servidor como, por ejemplo, acceder a bases de datos o llamar a beans de Enterprise Java.