

EJERCICIOS

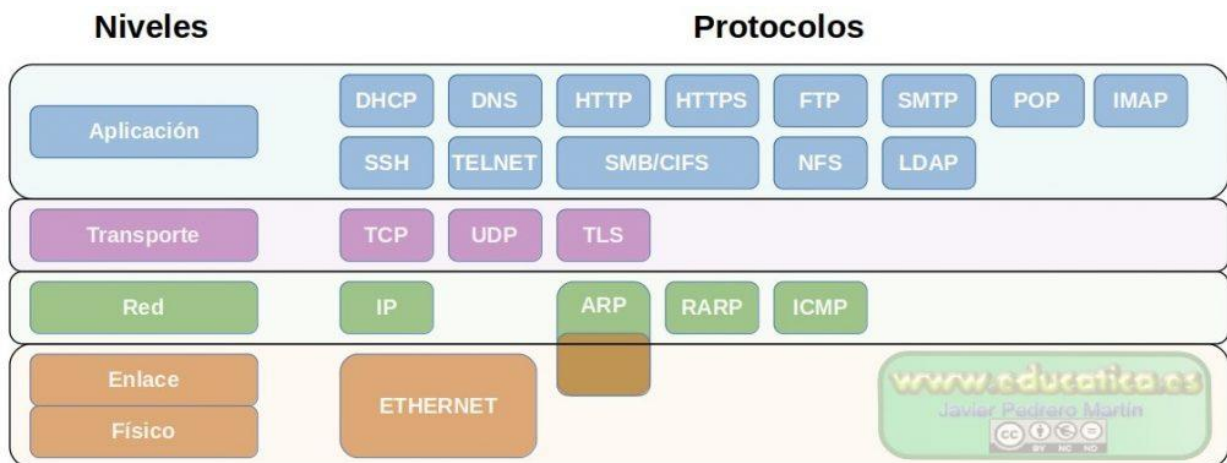
1. Protocolos de comunicaciones: IP, TCP, HTTP, HTTPS.

IP: Protocolo de la capa de Internet no orientado a conexión, basado en la transmisión de datos en forma de paquete, aunque este protocolo no asegura la recepción de los paquetes, por lo que **no es fiable**.

TCP: Protocolo de la capa de transporte orientado a conexión, es una forma segura de intercambio de datos, al requerir de la conexión y autorización entre el emisor y el receptor, cualquier pérdida de datos que se detecte, se soluciona, por lo que se considera un protocolo **fiable**.

HTTP: Protocolo de la capa de aplicación, significa Protocolo de Transferencia de Hipertexto. Tiene estructura cliente-servidor, esto quiere decir que una petición de datos es iniciada por el elemento que recibirá los datos (el cliente), normalmente un navegador Web, cada petición individual se envía a un servidor, el cual la gestiona y responde. (Puerto 80)

HTTPS: Protocolo de la capa de aplicación, significa Protocolo Seguro de Transferencia de Hipertexto, está basado en HTTP pero está destinado a la transferencia **segura** de datos, consigue esto mediante un cifrado basado en SSL/TLS con el fin de crear un canal cifrado entre el cliente y el servidor. (Puerto 443)



(Versión 1, Cristina Manjon)

2. Modelo de comunicaciones cliente – servidor y su relación con las aplicaciones web.

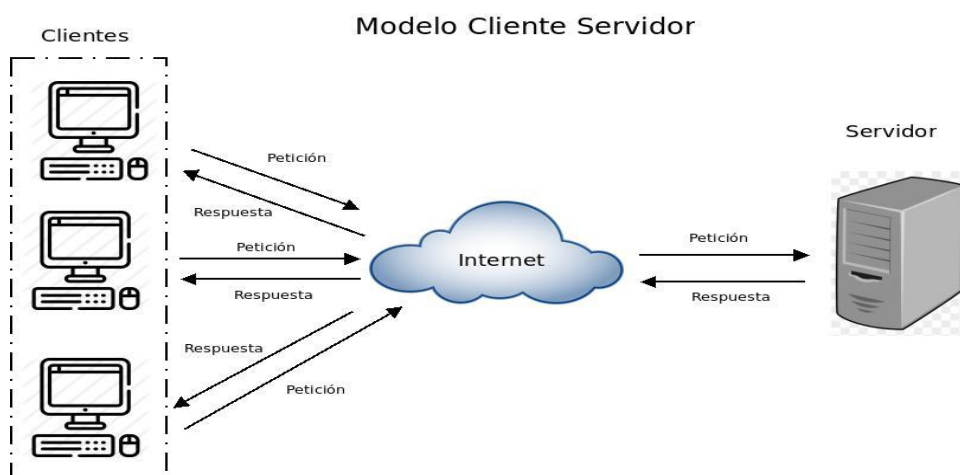
La arquitectura cliente-servidor es un modelo de diseño de software con dos partes claramente diferenciadas, el cliente y el servidor, que se reparten las tareas.

En este modelo de comunicaciones, un cliente realiza una petición al servidor, que la procesa y envía una respuesta. Este tipo de arquitectura nos permite conectar a varios clientes a los servicios que provee uno o varios servidores, lo cual es imprescindible para la mayoría de las aplicaciones y servicios web hoy en día.

¿Cuál es su relación con las aplicaciones web?

Cualquier aplicación web que queramos desarrollar va a necesitar utilizar un servidor que haga uso del protocolo http o https, y un cliente que realice la petición, es decir, el modelo cliente-servidor.

El cliente (Navegador) hace una petición de una página al servidor web. Este servidor web tiene código programado en un lenguaje de programación (PHP). El servidor procesa este documento y el resultado es la devolución en formato HTML al cliente.



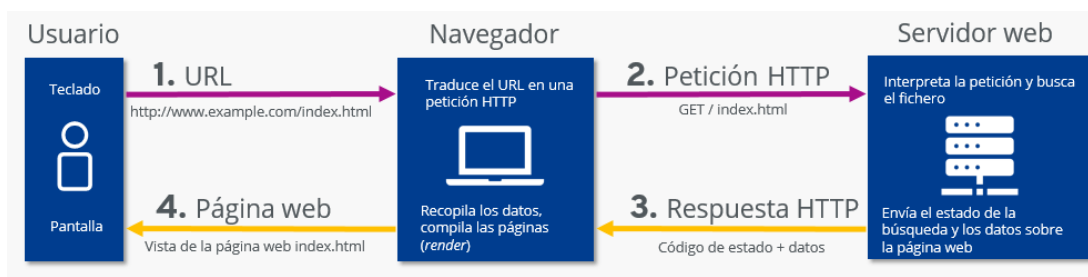
(version 1: Sonia Antón Llanes)

3. Estudio sobre los métodos de petición HTTP /HTTPS más utilizados.

- El método **GET** se emplea para adquirir un recurso ubicado en un servidor web, que devuelve los metadatos y el recurso que se precisa.
- El método **HEAD** realiza una función similar a la anterior, pero sólo se solicitan los metadatos y no el recurso en sí.
- El método **POST** se usa para enviar información a un recurso web del servidor, no para crear un elemento nuevo.
- El método **PUT** crea un recurso nuevo en el servidor o si éste ya existe lo reemplaza.
- El método **DELETE** solicita borrar el recurso especificado.

- El método **TRACE** sirve para el diagnóstico de errores o detección de servidores intermedios en la conexión mediante la monitorización de mensajes.
- El método **CONNECT** solicita saber si se tiene acceso a un host.
- El método **PATCH** solicita sobrescribir completamente un recurso. Se utiliza para actualizar, de manera parcial, una o varias partes.
- El método **OPTIONS** es utilizado para describir las opciones de comunicación para el recurso de destino.

GET	Solicita al servidor un objeto, recurso o documento
POST	Envía al servidor un documento
PUT	Solicita al servidor que haga accesible el documento enviado en una URI específica
DELETE	Solicita al servidor que elimine un documento
TRACE	Solicita una copia de la petición enviada
HEAD	Solicita al servidor el encabezado del recurso
OPTIONS	Solicita los métodos HTTP soportados por el servidor
CONNECT	Pregunta si se tiene acceso al servidor



(Versión

1: Bea Merino)

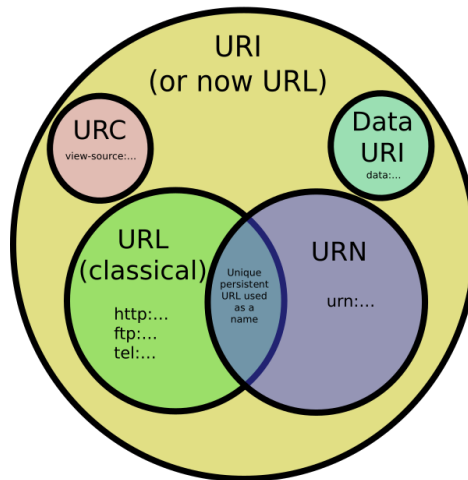
(version 2: Sonia Antón Llanes)

4. Estudio sobre el concepto de URI (Identificador de Recursos Uniforme) /URL/URN, estructura, utilidad y relación con el protocolo HTTP/HTTPS.

Cuando hablamos de URI, URL y URN hablamos de unas cadenas de texto, utilizadas como identificador o dirección de un recurso disponible en Internet.

El término URI engloba tanto a las URLs como a las URNs con lo que una URL es un tipo de URI y un URN también es un tipo de URI.

Cada vez que escribimos una dirección web en el navegador y accedemos a un sitio, estamos utilizando una URI.



URI

Uniform Resource Identifier: identificador uniforme de recursos.

Es una cadena que indica un recurso en internet, puede llevar datos adicionales que indiquen características como de qué forma acceder al recurso.

URL

Uniform Resource Locator: localizador uniforme de recursos. Se trata de una URI o cadena que indica un recurso en Internet para poder localizarlo.

URN

Uniform Resource Name: nombre uniforme de recursos.

Indica el nombre de un recurso que debe ser único en todo internet, además de independiente de su localización. No implica la disponibilidad del recurso.

(Versión 1: Ignacio Torío Álvarez)

5. Modelo de desarrollo de aplicaciones multicapa – comunicación entre capas – componentes – funcionalidad de cada capa.

El modelo de desarrollo de aplicaciones multicapa es un modelo donde las partes que componen el software se separan, dando lugar a diferentes capas.

Cada una de esas capas llevará a cabo una función determinada del sistema web, comunicándose entre ellas para asegurar el funcionamiento de la aplicación web. Las tres capas básicas serían:

- Capa de presentación: esta capa es la que ve el usuario. La programación de esta capa se centra en la información enviada por el servidor y en la captura de las acciones realizadas por el cliente.
- Capa de negocio: es la capa que conoce y gestiona las funcionalidades que esperamos del sistema o aplicación web. Es donde se reciben las peticiones del usuario y desde donde se envían las respuestas apropiadas tras el procesamiento de la información que ha proporcionado el cliente. Puede ser programada tanto en el entorno cliente como en el entorno servidor.
- Capa de persistencia o de datos: es la capa donde residen los datos y la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos, realizando el proceso de administración de datos, recibiendo solicitudes de almacenamiento o recuperando información desde la capa de negocio.

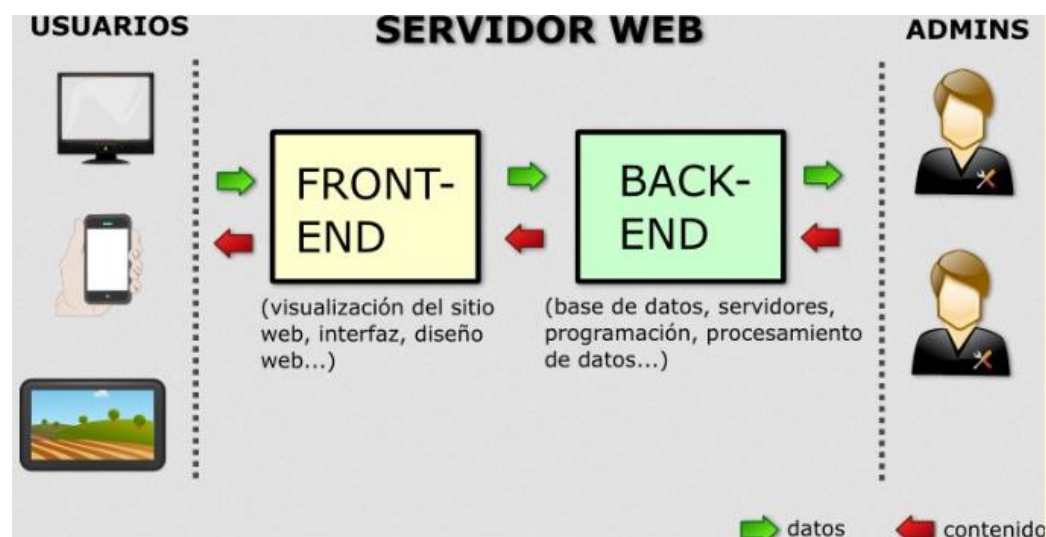
MVC vs MVP vs MVVM vs ... - Arquitecturas de diseño multicapa

(Versión 1: Elena de Antón)

6. Modelo de división funcional front-end / back-end para aplicaciones web.

Front-end: es el conjunto de páginas que ven la gran mayoría de usuarios que las usan (usuarios externos).

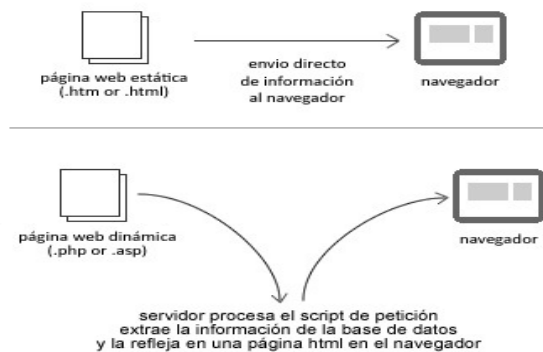
Back-end: que es otro conjunto de páginas dinámicas que utilizan las personas que producen el contenido y las que administran la aplicación web (usuarios internos o admins) para crear contenido, organizarlo, decidir la apariencia externa, etc.



(Versión 1, Cristina Manjon)

7. Página web estática – página web dinámica – aplicación web – mashup .

Una **página web** es un documento al que se puede acceder a través de un navegador (la información que muestra sólo se puede leer, no interactuar con ella).



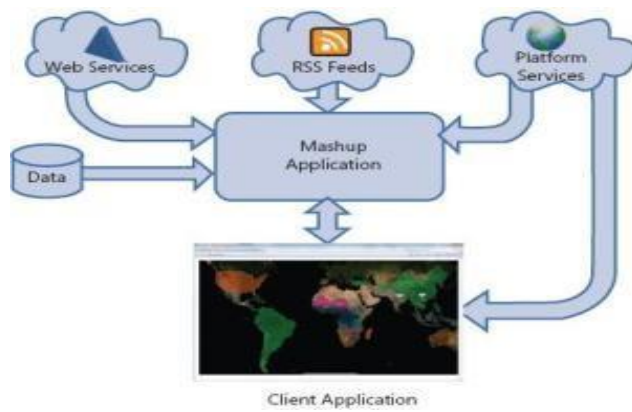
Una **página web estática** es aquella enfocada principalmente a mostrar una información permanente. Son funcionalmente básicas, con poca programación, principalmente en lenguajes HTML, CSS o JavaScript

Una **página web dinámica** ofrece una mayor interactividad con los usuarios que la visitan, permite la creación de aplicaciones dentro de la página web. Su creación es más compleja ya que tiene lenguajes de programación y gestión de bases de datos.

Características: página web estática vs página web dinámica	
<ul style="list-style-type: none">• Ausencia de movimiento y funcionalidades• Realizadas en XHTML o HTML• Proceso de actualización lento y manual• Hay que acceder al servidor para realizar cambios	<ul style="list-style-type: none">• Infinitas posibilidades de diseño y desarrollo• Se utilizan varias técnicas de programación• El proceso de actualización es sencillo• Permite muchas funcionalidades: bases de datos, foros...• El visitante puede alterar el diseño, contenidos...

<https://brandmedia.es/diferencias-pagina-web-estatica-dinamica-mejor/>

Una **aplicación web** es una página web dinámica, que tiene información sobre la que se puede interactuar e incluso cambiar, en otras palabras, son plataformas interactivas que se centran en que los usuarios realicen acciones.



Una **mashup**, o aplicación web híbrida, es una forma de integración y reutilización, es decir, es una aplicación web que integra o utiliza la información de una o varias páginas web y servicios de la red en una sola para ofrecer algo nuevo.

<https://sg.com.mx/content/view/256>

(version 1: Sonia Antón Llanes)

8. Componentes de una aplicación web.

- **Servidor web:** Generalmente suele ser Apache, se encarga de la comunicación a través de la red con el navegador del usuario.
- **Sistema gestor de bases de datos:** Se encarga de organizar toda la información con la que se trabaja, las bases de datos más empleadas son las relaciones, especialmente MySQL.
- **Módulo encargado de ejecutar o interpretar el código:** Se encarga de interpretar los comandos en el lenguaje utilizados (de Php, Asp, Perl, Python, Ruby) en la aplicación para procesarlos antes de enviar la página HTML al cliente.
- **Código desarrollado:**

Lenguajes de marcas: HTML, XML, CSS, XSL.

Lenguajes interpretados: PHP, JSP, Perl, Python.

Lenguajes compilados a código nativo: Aplicaciones CGI (C, C++).

Lenguajes compilados a código intermedio: Java EE (Servlets, JSP, EJB). ASP.net.

Lenguajes ejecutados en el lado del cliente: JavaScript.

Lenguajes de consulta: SQL.

(Versión 1: Bea Merino)

9. Programas ejecutados en el lado del cliente y programas ejecutados en el lado del servidor - lenguajes de programación utilizados en cada caso.

- Programas ejecutados en el lado del cliente:
 - Navegadores: Google Chrome, Firefox, Safari, Opera, Microsoft Edge.
- Programas ejecutados en el lado del servidor:
 - Servidor FTP: Filezilla Server.
 - Servidor SMTP: ¿?
- Lenguajes de programación del lado del cliente:
 - JavaScript.
- Lenguajes de programación del lado del servidor:
 - PHP.
 - JavaScript.
 - JSP (Java Server Pages).
 - ASP.Net (Active Server Pages).
 - Perl.

- o Python.
- o RUBY.
- o C, C#, C++ .
- o Java.
- o PL/SQL
- Lenguajes de marcas del lado del cliente:
 - o HTML
 - o CSS
 - o XML
- Lenguajes de definición de estructuras de datos
 - o XML
 - o JSON

(Versión 1: Ignacio Torío Álvarez)

10. Lenguajes de programación utilizados en el lado servidor de una aplicación web (características y grado de implantación actual).

Principales lenguajes de programación utilizados en el lado servidor de una aplicación web:

- PHP
 - Interpretado
 - Permite la programación orientada a objetos
 - Uno de los lenguajes de programación del lado del servidor más utilizados en la actualidad.
 - Lenguaje de código abierto, gratuito e independiente.
 - Tiene gran cantidad de documentación en su página web oficial.
 - Orientado al manejo de aplicaciones web dinámicas.
 - El código no es visible en el navegador web.
 - Tiene manejo de excepciones.
 - Multicapa.
- PERL
 - Lenguaje de programación en herramientas UNIX.
 - Disponible en gran selección de plataformas y sistemas operativos.
 - Desarrollo de aplicaciones muy rápido.
 - Lento para aplicaciones de bajo nivel.
- ASP

- Lenguaje comercializado por Microsoft
 - Muy utilizado para la creación y el desarrollo de sitios web dinámicos.
 - Crea aplicaciones tanto del lado del cliente como del servidor.
 - Se puede ejecutar en todos los navegadores del mercado.
 - Control de usuario personalizado.
 - División entre la capa de diseño y de código.
-
- JAVA
 - Ofrece la funcionalidad de un lenguaje potente, derivado de C y C++, pero sin las características menos usadas.
 - Orientado a objetos
 - Es distribuido, proporciona una gran biblioteca estándar y herramientas para que los programas puedan ser distribuidos.
 - Los programas son portables.
 - Es multihilo, lleva a cabo varias tareas simultáneamente. Esto permite mejorar el rendimiento y la velocidad de ejecución.
-
- PYTHON
 - Es un lenguaje interpretado, no compilado, usa tipado dinámico, fuertemente tipado.
 - Es multiplataforma, lo cual es ventajoso para hacer ejecutable su código fuente entre varios sistemas operativos.
 - Es un lenguaje de programación multiparadigma, el cual soporta varios paradigmas de programación como orientación a objetos, estructurada, programación imperativa y, en menor medida, programación funcional.
-
- JSP(Java ServerPages)
 - No debemos confundirlo con Java. Son lenguajes distintos.
 - Conjunta el poder de Java en el servidor y la flexibilidad de HTML en el browser.
 - No sólo se puede utilizar HTML, sino también XML o WML.
 - Hace más fácil reusar componentes con JavaBeans los cuales realizan tareas más específicas.
 - Su función es saber como procesar una solicitud para crear una respuesta.
 - Soporta contenido dinámico que refleja las condiciones del mundo real.
 - Es más rápido y fácil crear aplicaciones de web
 - Capaz de instanciar cualquier clase de Java

Buscar ejemplos de aplicaciones web que los utilicen

Revisar y mejorar esta lista

(Versión 2: Elena de Antón, Cristina Manjon (añado Java, Phyton y JSP))

11. Características y posibilidades de desarrollo de una plataforma XAMPP.

XAMPP es un servidor independiente de plataforma de código libre. Te permite instalar de forma sencilla **Apache** en tu propio ordenador, sin importar tu sistema operativo (Linux, Windows, MAC o Solaris). Y lo mejor es que su uso es gratuito.

XAMPP incluye además **servidores de bases de datos** como MySQL y SQLite con sus respectivos gestores phpMyAdmin y phpSQLiteAdmin. Incorpora también el **intérprete de PHP**, el intérprete de Perl, **servidores de TTP** como ProTTPD ó TileZilla TTP Serve, etc. entre muchas cosas más.

Sin embargo, es bueno acotar que la seguridad de datos no es su punto fuerte, por lo cual no es suficientemente seguro para ambientes grandes o de producción.



Buscar un tutorial de instalación en una máquina windows y poner el enlace

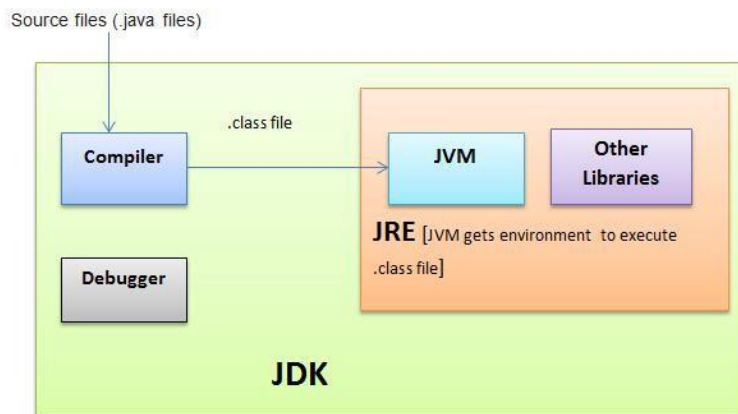
(Versión 1, Cristina Manjon)

12. En qué casos es necesaria la instalación de la máquina virtual Java (JVM) y el software JDK en el entorno de desarrollo y en el entorno de explotación.

El software JDK lo utilizamos cuando estamos realizando una aplicación que contenga código de Java. Este software se encarga de convertir nuestro código fuente (.java) en bytecode (.class), el cual puede ser interpretado y ejecutado por la máquina virtual de java.

El software JDK es necesario en el entorno de desarrollo.

La instalación de la máquina virtual Java es necesaria cuando se va a **ejecutar** el código precompilado de Java, es decir, lo utilizamos para el entorno de explotación.



Tutorial de instalación de JVM en una máquina windows

https://www.agenciatributaria.es/AEAT.internet/Inicio/Ayuda/_comp_Consultas_informaticas/Categorias/Configuracion_de_navegadores_y_otros_requisitos/Maquina_Virtual_Java/_Descarga_e_instalacion_de_la_Maquina_Virtual_Java/_Descarga_e_instalacion_de_la_Maquina_Virtual_Java/_Descarga_e_instalacion_de_la_Maquina_Virtual_Java.shtml

[!\[\]\(8bba887393ca45b761e5cb49e755e762_img.jpg\) Cómo DESCARGAR Java Virtual Machine](#) [!\[\]\(b898b980f2d860cdb0237afbc3664529_img.jpg\) Instalar Java Virtual Machine para Windows 10](#)



(version 1: Sonia Antón Llanes)

13. IDE más utilizados (características y grado de implantación actual).

Eclipse

Es uno de los entornos más conocidos y utilizados por los programadores, ya que se trata de un entorno de programación de código abierto, multiplataforma y multilenguaje.

Está en constante evolución y hay muchos tutoriales por la red.

NetBeans

Entorno multilenguaje y multiplataforma.

El lenguaje que mejor soporta es Java, ya que fue creado por Oracle y su creación fue para ser el IDE de Java.

Visual Studio

Diseñado por Microsoft y es uno de los mejores entornos de programación que existen, es multilenguaje y multiplataforma.

Su versión gratuita es **Visual Studio Community**, muy parecido al Visual Studio de pago, pero con menos herramientas y distintas licencias.

<https://www.diarlu.com/entornos-de-desarrollo-integrado/>

(Versión 1: Bea Merino)

14. Servidores HTTP /HTTPS más utilizados (características y grado de implantación actual).

Poner un enlace a la web del fabricante

- Apache HTTPS

CARACTERÍSTICAS
<ul style="list-style-type: none">- Es gratuito- Consume muchos recursos y no es muy eficiente si hay mucho tráfico- Es fácil de configurar y usar- Es bastante seguro y estable- Funciona con Windows y Linux- Funciona de inmediato con los sitios de WordPress- Mayor selección de módulos y una comunidad muy grande

- Nginx:

CARACTERÍSTICAS
<ul style="list-style-type: none"> - Multifunción: Puede actuar como un servidor proxy inverso web y de correo para los protocolos (TCP, UDP, HTTP, HTTPS, SMTP, POP3 e IMAP) equilibrador de carga y una caché HTTP. - Configura un proceso de trabajo por CPU y no por cada petición de un equipo. - Consume menos recursos y memoria - Maneja muchas conexiones con poca memoria - Mejor escalabilidad - Recomendado para contenido estático - Recomendado para sitios con VPS - Tiene una comunidad pequeña y menos módulos y complementos

- Microsoft IIS (Internet Information Service):

CARACTERÍSTICAS
<ul style="list-style-type: none"> - Servidor web de pasgo creado por Microsoft para sus sistemas operativos Windows. - Se integra con Active Directory por lo que es muy útil para una intranet en la que haya que controlar a usuarios que se conectan a servicios web externos e internos. - Consume muchos recursos, pero es muy potente y muy personalizable. - Soporte personalizado. - Empleado sobre todo para apps desarrolladas con ASP/ASP.NET - Al ser exclusivo de Windows por lo que no se integra bien con otras tecnologías y otros servidores.

- GWS (Google Web Server):

CARACTERÍSTICAS
<ul style="list-style-type: none"> - Servidor web privado escrito en C++ por Google basado en Linux. - Gestionable desde cualquier lugar al ofrecer un entorno sin servidor. - Permite conectar servicios en la nube basados en la arquitectura de microservicios. - Ofrece análisis de datos a gran escala, Big Data y Machine Learning. - De pago, escalable y con muchos productos según pagues más.

Gráfico:

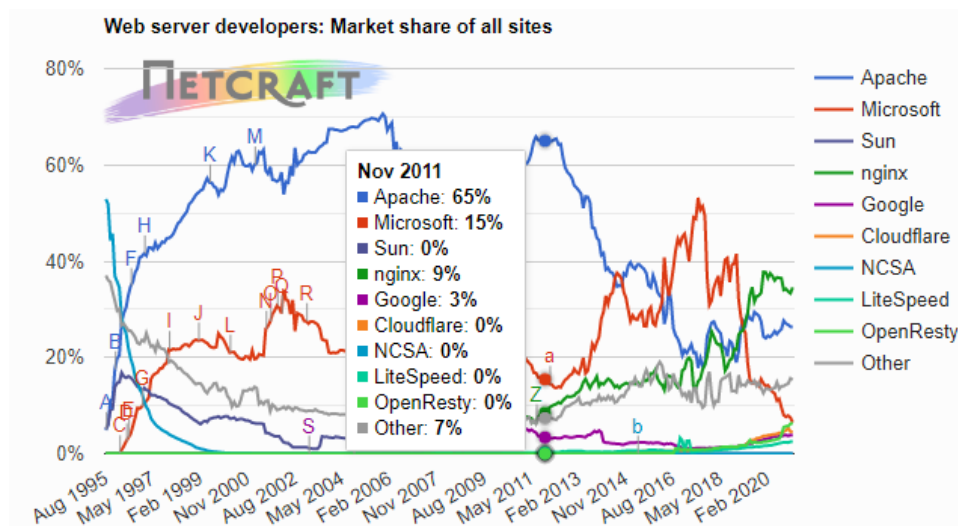


Tabla de información [aquí](#).

Comparación de todos los servidores web: [aquí](#).

Servidores web Wikipedia: [aquí](#).

(Versión 1: Ignacio Torío Álvarez)

15. Apache HTTP vs Apache Tomcat

Apache Server y Apache Tomcat son dos de los productos desarrollados por Apache Software Foundation.

Apache es un servidor web HTTP, mientras que Tomcat es un entorno de contenedor de Servlet. Sin embargo, el servidor Tomcat viene con su propio componente de servidor HTTP.

Apache HTTP está optimizado para manejar contenido web estático y dinámico, y carece de la capacidad de administrar Servlets Java y JSP. Mientras que Tomcat está orientado al contenido en Java y se encarga del JSP.

Añadir un gráfico que represente la colaboración entre estos servidores

(Version 1: Elena de Antón)

16. Navegadores HTTP /HTTPS más utilizados (características y grado de implantación actual).

Google Chrome: Creado por google, es uno de los más conocidos y usados, básicamente porque es el que asegura una velocidad mayor. Está diseñado para ofrecer una mayor seguridad.

Mozilla Firefox: aparte de ser uno de los más veteranos (salió en el año 2003) es sólido, estable y presenta muy pocos errores. Es el segundo navegador más utilizado. Además, es un navegador altamente personalizable.











Opera: Es el navegador web alternativo por excelencia. Ha sido de los más utilizados en los teléfonos móviles, hasta la popularización de los smartphones.

Safari: navegador web asociado a los Macs, Safari no es el navegador más rápido de todos los que existen, pero es estable y eficiente.

Internet Explorer: Explorer mantiene su amplia cuota de mercado (alrededor del 60-65%) gracias a que fue el primero en salir y también que viene predeterminado en todos los PCs de Microsoft. Es el navegador que presenta más problemas de seguridad de todos los candidatos.

Browser Statistics March 2021

Here you see the market shares of the most used browsers for the current month.

Rank	Browser	
1	 Google Chrome	44.61% 
2	 Safari	34.24% 
3	 Firefox	11.67% 
4	 Edge	6.63% 
5	 Mozilla	2.85% 

CUADRO COMPARATIVO DE LOS NAVEGADORES DE INTERNET

2006

Navegador	Características	Usabilidad	Sistema operativo
Internet Explorer	<ul style="list-style-type: none"> -Ha sido el navegador más utilizado -Sus versiones más recientes son la 8.0 y la 9.0 -Sustituyó a Netscape -Guarda información temporal para tener una mayor accesibilidad -Exploración de pestañas -Recolección de errores 	<ul style="list-style-type: none"> -Es una interfaz de usuario de FTP -Utiliza una navegación con pestañas 	Microsoft Windows Windows Update
Opera	<ul style="list-style-type: none"> -Realiza múltiples tareas -Se puede ver en todos los navegadores -Idiomas soportados -Papelería de pestañas cerradas -Recuerda el contenido de las pestañas -Barra de búsqueda de tamaños ajustables 	<ul style="list-style-type: none"> - Es uno de los más utilizados -tiene una navegación rápida 	Microsoft Windows
Google chrome	<ul style="list-style-type: none"> -Tiene seguridad, velocidad y estabilidad -Descarga periódicamente actualizaciones -Nos muestra cuando las paginas ponen en riesgo a nuestro equipo - Incluye opciones para ir atrás, adelante, recargar página, ir y cancelar -Tienen pestañas miniaturas de las paginas utilizadas anteriormente 	<ul style="list-style-type: none"> -Los puedes descargar fácilmente 	Microsoft Windows Mac OS X Linux
Mozilla	<ul style="list-style-type: none"> -Utiliza criptografía para una mayor seguridad -Su navegación se basa en pestañas -Se le pueden agregar pestañas -Tiene un sistema propio de extensiones -Tiene una identidad visual 	<ul style="list-style-type: none"> -Es fácil su accesibilidad - Cual quiera lo puede obtener 	Microsoft Windows Mac OS X
Safari	<ul style="list-style-type: none"> -Tiene una navegación por medio de pestañas -Corrector ortográfico -Búsqueda progresiva -Atajos del teclado -Su entorno es para desarrollos web's 	<ul style="list-style-type: none"> -casi ya no lo utilizan pues a sido sustituido por los navegadores modernos 	Mac OS X iOS Microsoft Windows.
iCab	<ul style="list-style-type: none"> -Se pueden filtrar imágenes -Se camufla gea completamente como otro navegador 		Mac OS

(Versión 1, Cristina Manjon)

17. Generadores de documentación HTML

(PHPDoc): PHPDocumentor, **Doxygen**, ApiGen, ...

A medida que crece una aplicación se va complicando el manejo del proyecto para el desarrollador y los colaboradores. Para analizar y/o entender las partes del proyecto sin necesidad de analizar el código línea a línea, el estándar de programación define unas reglas para escribir y documentar código.

Un **generador de documentación** es una herramienta de programación que genera documentación destinada a los programadores (documentación de API) o a usuarios finales, o a ambos, a partir de un conjunto de código fuente especialmente documentado.

PHPDoc es una adaptación de javadoc para PHP que define un estándar oficial para comentar código php. Ofrece tres ventajas:

- Hace comentarios que puede leerse en un método estándar, para comentar aspectos de código que normalmente se ignoran.
- Permite a los generadores de documentación, como phpDocumentor, crear la documentación API en buen formato y fácil de entender.
- Permite a algunos IDEs (NetBeans, Aptana, Zend y PhpStorm) que interpreten los tipos de variables y otras ambigüedades del lenguaje de programación.

<https://es.wikipedia.org/wiki/PHPDoc>

Doxygen es un generador de documentación para C++, C, Java, Objective-C, Python, IDL (versiones Corba y Microsoft), VHDL y en cierta medida para PHP, C# y D. Dado que es fácilmente adaptable, funciona en la mayoría de sistemas Unix así como en Windows y Mac OS X. La mayor parte del código de Doxygen está escrita por Dimitri van Heesch.

<https://es.wikipedia.org/wiki/Doxygen>

PHPDocumentor es un generador de documentación de código abierto escrito en PHP. Automáticamente analiza el código fuente PHP y produce la API de lectura y documentación del código fuente. Genera esta documentación en base al estándar formal PHPDoc.

<https://es.wikipedia.org/wiki/PhpDocumentor>

ApiGen otro generador de documentación API profesional a partir del código fuente PHP, similar a phpDocumentor o phpDoc. ApiGen tiene soporte para espacios de nombres PHP 5.3, paquetes, vinculación entre documentación, referencias cruzadas a clases estándar PHP y documentación general, creación de código fuente resaltado y soporte experimental para rasgos PHP 5.4

(version 1: Sonia Antón Llanes)

18. Repositorios de software – sistemas de control de versiones: GIT, CVS, Subversion

Un repositorio de software es un lugar de almacenamiento del cual pueden ser recuperados e instalados los paquetes de software en un ordenador.

- **GIT**: es un software de **control de versiones** diseñado por Linus Torvalds. Su propósito es **registrar los cambios en archivos del ordenador y coordinar el trabajo que varias personas realizan sobre archivos compartidos**.

<https://docs.github.com/es/github>

- **CVS**: Concurrent Versions System, es una aplicación informática que implementa un sistema de **control de versiones: mantiene el registro de todo el trabajo y los cambios en los ficheros** (código fuente principalmente), que forman un proyecto **y permite que distintos desarrolladores colaboren**, aunque no estén juntos.
- **Subversion**: Apache Subversion es una herramienta de **control de versiones** de código abierto basada en un repositorio cuyo funcionamiento se asemeja enormemente al de un sistema de ficheros. **Utiliza el concepto de revisión para guardar los cambios producidos en el repositorio**. Entre dos revisiones **sólo guarda el conjunto de modificaciones** (delta), optimizando así al máximo el uso de espacio en disco.

(Versión 1: Bea Merino)

19. Propuesta de configuración del entorno de desarrollo para la asignatura de Desarrollo web del lado servidor en este curso (incluyendo las versiones): xxx-USED y xxx-WXED.

- Configuración del cliente del entorno de desarrollo **cliente windows**:

Sistema operativo	Windows 10 Pro x64 bits (versión 20H2)		
Memoria RAM	4 GB		
Particiones	Tamaño del disco: 500 GB		
	Capacidad	Tipo	Montaje
	500 MB	NTFS	recovery
	100 GB	NTFS	c:/ (sistema)
	400 GB	NTFS	d:/ (datos)
Nombre del equipo	ITA-W10ED-106		
Grupo de trabajo	daw206.local		

<i>Usuarios</i>	<ul style="list-style-type: none"> - Administrador (admin por defecto), - myadmin/P@ssw0rd (administrador del equipo), - usuario/P@ssw0rd (usuario del equipo),
<i>Programas</i>	<ul style="list-style-type: none"> - Google Chrome, Mozilla Firefox, Opera, Microsoft Edge (navegadores) - FileZilla Client (cliente FTP). - Thunderbird (gestor de correo). - Libreoffice (editor de documentos). - Notepad ++, Sublime Text (editores de texto). - PuTTY (conexión SSH). - Eclipse, Netbeans, Visual Studio Code (entornos de desarrollo). - 7-Zip (descompresor de archivos). - JDK (kit de desarrollo de java). - PDF Viewer (visor de documentos). - GIT Bash (repositorio).

- Configuración del servidor del entorno de desarrollo **servidor Ubuntu server:**

Sistema operativo	Ubuntu Server x64 bits (versión 20.04 LTS)																											
Memoria RAM	1 GB																											
Particiones	<table><tr><th colspan="3">Tamaño del disco: 500 GB</th></tr><tr><th>Capacidad</th><th>Tipo</th><th>Montaje</th></tr><tr><td>2 GB</td><td>bios/grub</td><td></td></tr><tr><td>2 GB</td><td>ext4</td><td>/boot</td></tr><tr><td>100 GB</td><td>ext4</td><td>/</td></tr><tr><td>50 GB</td><td>ext4</td><td>/var</td></tr><tr><td>50 GB</td><td>Ext4</td><td>/usr</td></tr><tr><td>290 GB</td><td>ext4</td><td>/home</td></tr><tr><td>4 GB</td><td>swap</td><td></td></tr></table>	Tamaño del disco: 500 GB			Capacidad	Tipo	Montaje	2 GB	bios/grub		2 GB	ext4	/boot	100 GB	ext4	/	50 GB	ext4	/var	50 GB	Ext4	/usr	290 GB	ext4	/home	4 GB	swap	
Tamaño del disco: 500 GB																												
Capacidad	Tipo	Montaje																										
2 GB	bios/grub																											
2 GB	ext4	/boot																										
100 GB	ext4	/																										
50 GB	ext4	/var																										
50 GB	Ext4	/usr																										
290 GB	ext4	/home																										
4 GB	swap																											
Nombre del equipo	ITA-USED-206																											
Grupo de trabajo	daw206.local																											
Usuarios	<ul style="list-style-type: none">- root/P@ssw0rd (super admin).- myadmin/P@ssw0rd (administrador del equipo).- usuario/P@ssw0rd (usuario del equipo).- operadorweb/P@ssw0rd (gestor de la página web).- operadordb/P@ssw0rd (gestor de la base de datos).																											
Programas	<ul style="list-style-type: none">- Apache (servidor web HTTP).- PHP (lenguaje de programación).- MySQL (gestor de bases de datos).																											

- SSH (conexión segura y acceso remoto al servidor).
- Xdebug (depuración de código).

- Servidor de repositorio en la nube [GitHub](#)

(Versión 1: Ignacio Torío Álvarez)

20. Propuesta de configuración del entorno de explotación para la asignatura de Desarrollo web del lado servidor en este curso (incluyendo las versiones): xxx-USEE.

Entorno de explotación Windows XXX-W10EE:

- ✓ Sistema operativo: Windows 10
- ✓ Navegador: Firefox 81.0
- ✓ ~~IDE: NetBeans 12.0~~
- ✓ ~~Servidor Web: Apache HTTP: Apache 2.4~~
- ✓ ~~Ofimática: Microsoft Office~~
- ✓ ~~Ciente SSH: Filezilla, Putty~~
- ✓ ~~Sistema gestor de bases de datos: MySQL 8.0~~
- ✓ ~~Repositorio: Github~~

Entorno de explotación Ubuntu Server XXX.USEE:

- ✓ Sistema operativo: Ubuntu Server 20.04
- ✓ Servidor Web: Apache HTTP: Apache 2.4
- ✓ Sistema gestor de bases de datos: MySQL
- ✓ Servidor administración remoto: SSH
- ✓ Servidor de transferencia de ficheros: SFTP(SSH)

(Versión 1: Elena de Antón)

21. Realizar un estudio sobre los siguientes conceptos y su relación con el desarrollo de aplicaciones web: CMS – Sistema de gestión de contenidos ERP – Sistema de planificación de los recursos empresariales

CMS – Sistema de gestión de contenidos: es “una herramienta software para crear, administrar y gestionar un sitio web”.

Sus 3 funciones principales son:

- Creación del sitio web.
- Gestión y mantenimiento del sitio web.
- Administración del sitio web y del propio CMS.



Todos ellos estan basados en PHP

ERP – Sistema de planificación de los recursos empresariales: es un “sistema de planificación de recursos empresariales”, estos programas **se hacen cargo de distintas operaciones internas de una empresa**, desde producción a distribución o incluso recursos humanos.

Las principales ventajas de estos sistemas son:

- Automatización de procesos de la empresa.
- Disponibilidad de la información de la empresa en una misma plataforma.
- Integración de las distintas bases de datos de una compañía en un solo programa.
- Ahorro de tiempo y costes.

Los sistemas ERP suponen una gran inversión para las empresas pero las empresas que adquieren un ERP notan un aumento la productividad.



Algunos ejemplos son:



[Pincha Aqui si quieres saber mas a fondo sus diferencias](#)

(Versión 1, Cristina Manjon)

22. Elegir y realizar un estudio y una presentación para la exposición del trabajo sobre una de las siguientes arquitecturas de desarrollo de Aplicaciones Web:

•MEAN (con MongoDB y con MySQL)

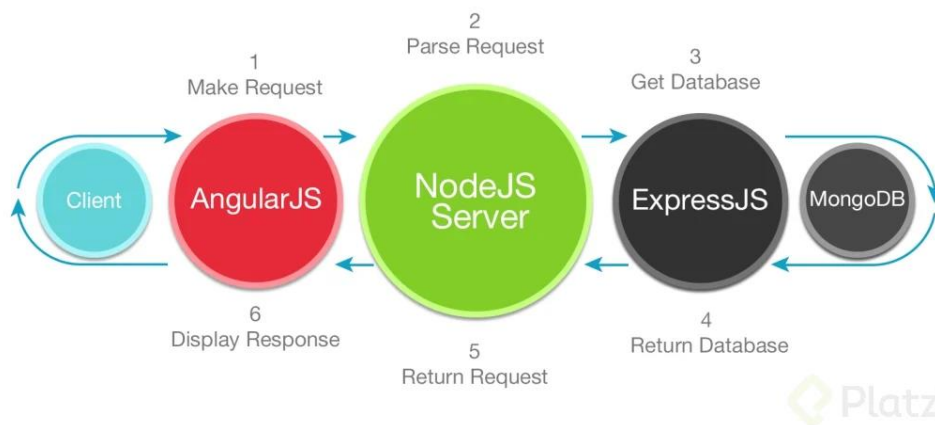
<https://es.wikipedia.org/wiki/MEAN>

<https://platzi.com/blog/que-es-mean-full-stack-javascript/>

MEAN es desarrollo full-stack en JavaScript, es decir, es **el conjunto de tecnologías necesarias para el desarrollo de todas las capas de una aplicación web con JavaScript**.

Está compuesto por cuatro de las más importantes tecnologías de la industria:

[MongoDB](#), [Express](#), [Angular](#) y [Node.js](#).



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, 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 afecte su funcionamiento.

ExpressJS: es un módulo de NodeJS , es decir, un framework para hacer aplicaciones web. Es rápido y fácil de usar.

En una arquitectura MVC – Model View Controller, Express es el Control, la parte de la aplicación que **te permite conectar todos los componentes, gestionar y procesar los datos y enviarlos a las capas superiores** que le permite al usuario tener acceso a la información.

AngularJS: es un framework que facilita la manipulación del DOM y por lo tanto 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. Angular también tiene la capacidad de hacer el enrutamiento del sistema y claro está también incorpora la

función de ajax (asincronico javascript y xml), para hacer peticiones al servidor sin tener que cargar toda la página.

NodeJS: (es la base de todo el conjunto) 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.

•Java EE vs Spring

<https://www.educba.com/java-ee-vs-spring/>

<https://openwebinars.net/blog/conoce-que-es-spring-framework-y-por-que-usarlo/>

Java Enterprise Edition es una plataforma de programación para desarrollar y ejecutar software de aplicaciones en el lenguaje de programación Java.

Spring es un framework para el desarrollo de aplicaciones y contenedor de inversión de control, de código abierto para la plataforma Java 2. Se puede considerar como el padre de los frameworks Java, ya que da soporte a otros frameworks.

Spring fue lanzado por primera vez en junio de 2003 bajo la licencia Apache 2.0, siendo una plataforma Java de código abierto. Convirtiéndose desde entonces en el framework más popular para Java empresarial, para crear código de alto rendimiento, liviano y reutilizable. Ya que su finalidad es estandarizar, agilizar, manejar y resolver los problemas que puedan ir surgiendo en el trayecto de la programación.

	Java EE	Spring
Arquitectura	Basado en un marco arquitectónico tridimensional, es decir, niveles lógicos, niveles de cliente y niveles de presentación.	Se basa en una arquitectura en capas que incluye muchos módulos. Estos módulos se fabrican en la parte superior de su contenedor principal.
Lenguaje	Utiliza un lenguaje orientado a objetos de alto nivel que tiene cierto estilo y sintaxis.	No tiene un modelo de programación determinado.
Interfaz	Por lo general, tiene una interfaz gráfica de usuario creada a partir de las API de Project Swing o Abstract Window Toolkit.	La sintaxis es la misma en todas partes, independientemente de un IDE o un compilador
Inyección de dependencia	Utiliza inyección de dependencia	Utiliza inyección de dependencia
Estructura	Puede estar basado en web o no	Basado en casi 20 módulos
Velocidad	Bastante buena velocidad	Spring es más lento que Java EE

•Microsoft .NET

<https://openwebinars.net/blog/que-es-net-framework/>

Microsoft .net es el conjunto de estructuras y tecnologías que proporciona Microsoft para una programación más sencilla orientada a las redes e internet, con independencia de la plataforma hardware utilizada. Para programar en .NET existen hoy en día **más de 20 de lenguajes de programación**, pero C# y Visual Basic son los más populares.

Los principales componentes del entorno de trabajo .Net recogen la **biblioteca de clases** base, los **lenguajes de programación** y el **entorno común** para ejecución de lenguajes (CLR).

• Angular 7:

Framework para aplicaciones web desarrollado en TypeScript, de código abierto, mantenido por Google, que se utiliza para crear y mantener aplicaciones web de una sola página. Su objetivo es aumentar las aplicaciones basadas en navegador con capacidad de Modelo Vista Controlador (MVC), en un esfuerzo para hacer que el desarrollo y las pruebas sean más fáciles.

La biblioteca lee el HTML que contiene atributos de las etiquetas personalizadas adicionales, entonces obedece a las directivas de los atributos personalizados, y une las piezas de entrada o salida de la página a un modelo representado por las variables estándar de JavaScript.

Angular se basa en clases tipo "Componentes", cuyas propiedades son las usadas para hacer el binding de los datos. En dichas clases tenemos propiedades (variables) y métodos (funciones a llamar).

Características

Velocidad y rendimiento

- Generación de código: Angular convierte tus plantillas en código altamente optimizado.
- Universal: Ejecuta la primera vista de tu aplicación en node.js, .NET, PHP, y otros servidores para renderizado de forma casi instantánea obteniendo solo HTML y CSS.

- División del código: Las aplicaciones de Angular se cargan rápidamente gracias al nuevo enrutador de componentes. Éste ofrece una división automática de códigos para que los usuarios sólo carguen el código necesario para procesar la vista que solicitan.

Productividad

- Plantillas: Permite crear rápidamente vistas de interfaz de usuario.
- Angular CLI: Las herramientas de línea de comandos permiten empezar a desarrollar rápidamente, añadir componentes y realizar test, así como previsualizar de forma instantánea la aplicación.

- IDEs: Obtén sugerencias de código inteligente, detección de errores y otros comentarios en la mayoría de los editores populares e IDEs.

Historia completa del desarrollo

- Testing: Utiliza Karma para realizar pruebas unitarias, y Protractor para realizar pruebas end-to-end de forma rápida y estable.
- Animación: Permite crear animaciones complejas y de alto rendimiento con muy poco código a través de la intuitiva API de Angular.
- Accesibilidad: Posee características para crear aplicaciones accesibles con los componentes disponibles para ARIA



Symfony

SYMFONY: Es un framework diseñado para desarrollar aplicaciones web basado en el patrón Modelo Vista Controlador. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación.



laravel

LARAVEL: Es un framework de código abierto para desarrollar aplicaciones y servicios web con PHP 5 y PHP 7. Su filosofía es desarrollar código PHP de forma simple, muy potente y tiene una interfaz elegante y divertida de usar, evitando el "código espagueti". La mayor parte de su estructura está formada por dependencias, especialmente de Symfony, lo que implica que el desarrollo de Laravel dependa también del desarrollo de sus dependencias.



CakePHP

CAKEPHP: CakePHP es un framework rápido para PHP, libre, de código abierto. Se trata de una estructura que sirve de base a los programadores para que éstos puedan crear aplicaciones Web. Creado sobre los conceptos de Ruby on Rails.



CodeIgniter

CODEIGNITER: Es un framework para el desarrollo de aplicaciones en php, que utiliza el MVC. Esto permite a los programadores o desarrolladores Web mejorar su forma de trabajar, además de dar una mayor velocidad a la hora de crear páginas Webs.

(Versión 1, Sonia , Bea y Cristina)