
TEMA 1-DESARROLLO WEB EN ENTORNO SERVIDOR

EJERCICIOS.....	2
1. Protocolos de comunicaciones: IP, TCP, HTTP, HTTPS.....	2
2. Modelo de comunicaciones cliente – servidor y su relación con las aplicaciones web. ...	4
3. Estudio sobre los métodos de petición HTTP /HTTPS más utilizados.....	5
4. Estudio sobre el concepto de URI (Identificador de Recursos Uniforme) /URL/URN, estructura, utilidad y relación con el protocolo HTTP/HTTPS.	6
5. Modelo de desarrollo de aplicaciones multicapa – comunicación entre capas – componentes – funcionalidad de cada capa.....	7
6. Modelo de división funcional front-end / back-end para aplicaciones web	8
7. Página web estática – página web dinámica – aplicación web – mashup.	9
8. Componentes de una aplicación web.	10
9. Programas ejecutados en el lado del cliente y programas ejecutados en el lado del servidor - lenguajes de programación utilizados en cada caso Lado del cliente.....	11
11. Características y posibilidades de desarrollo de una plataforma XAMPP.	13
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.	14
13. IDE más utilizados (características y grado de implantación actual).	15
14. Servidores HTTP /HTTPS más utilizados (características y grado de implantación actual).....	17
15. Apache HTTP vs Apache Tomcat.....	18
16. Navegadores HTTP /HTTPS más utilizados (características y grado de implantación actual).	19
17. Generadores de documentación HTML (PHPDoc): PHPDocumentor, ApiGen,	21

EJERCICIOS

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

Respuesta:

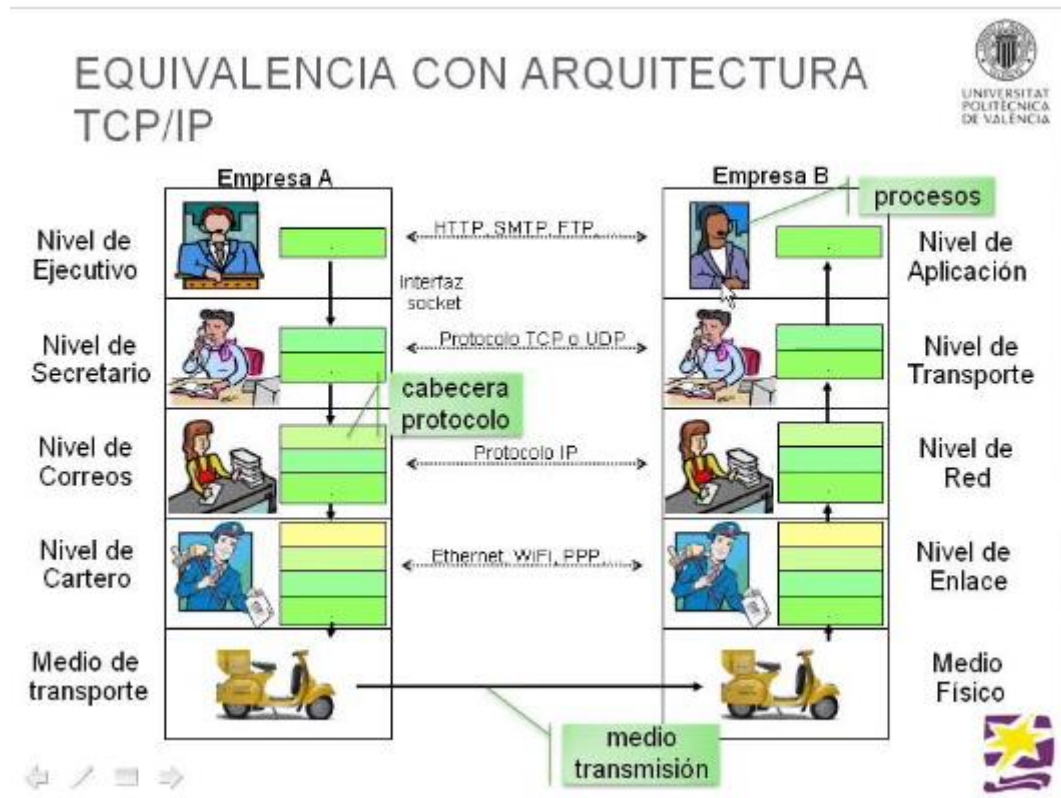
- IP: Es un protocolo para enrutar y direccionar paquetes que pueden viajar a través de la red y llegar al destino correcto.
- TCP: Es el protocolo encargado de proporcionar comunicación de un punto a otro entre dos hosts.
- HTTP: Es un protocolo de transferencia sobre el que se basa la www. Funciona como base para los intercambios de datos realizados en la web.
- HTTPS: Es un protocolo de transferencia de hipertexto seguro y es la versión encriptada de HTTP. Se utiliza para una comunicación segura a través de internet.

Enlaces:

- [IP](#)
- [TCP](#)
- [HTTP](#)
- [HTTPS](#)

Modelos/imágenes:





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

Respuesta:

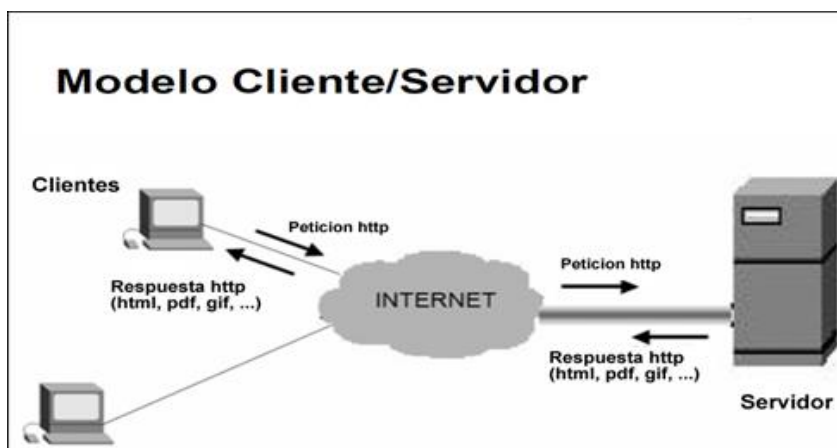
El modelo cliente servidor es un modelo de comunicación mediante un servidor y un cliente a través de solicitudes HTTP O HTTPS. Una aplicación consta de una parte de servidor y una de cliente, que se pueden ejecutar en el mismo o en diferentes sistemas.

Enlaces:

[Modelo cliente servidor en Wikipedia.](#)

[Modelo cliente servidor en IBM.](#)

Modelos/imágenes:



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

Respuesta:

- **GET:** Una petición GET solicita al servidor una información o recurso concreto.
- **POST:** El método POST se utiliza para enviar una entidad a un recurso en específico.
- **HEAD:** Entregara al cliente los encabezados de respuesta y solicitud.
- **PUT:** Se encarga de modificar/actualizar datos.
- **DELETE**
- **CONNECT**
- **OPTIONS**
- **TRACE**
- **PATCH**

Enlaces:

[Métodos de petición en Keepcoding.](#)

[Métodos de petición en Wikipedia.](#)

Modelos/imágenes:



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

Respuesta:

El URI sirve para acceder a un recurso físico o abstracto a través de Internet. El formato es el siguiente: “protocolo://servidor: puerto/directorio/pagina”. Utiliza el protocolo http o https.

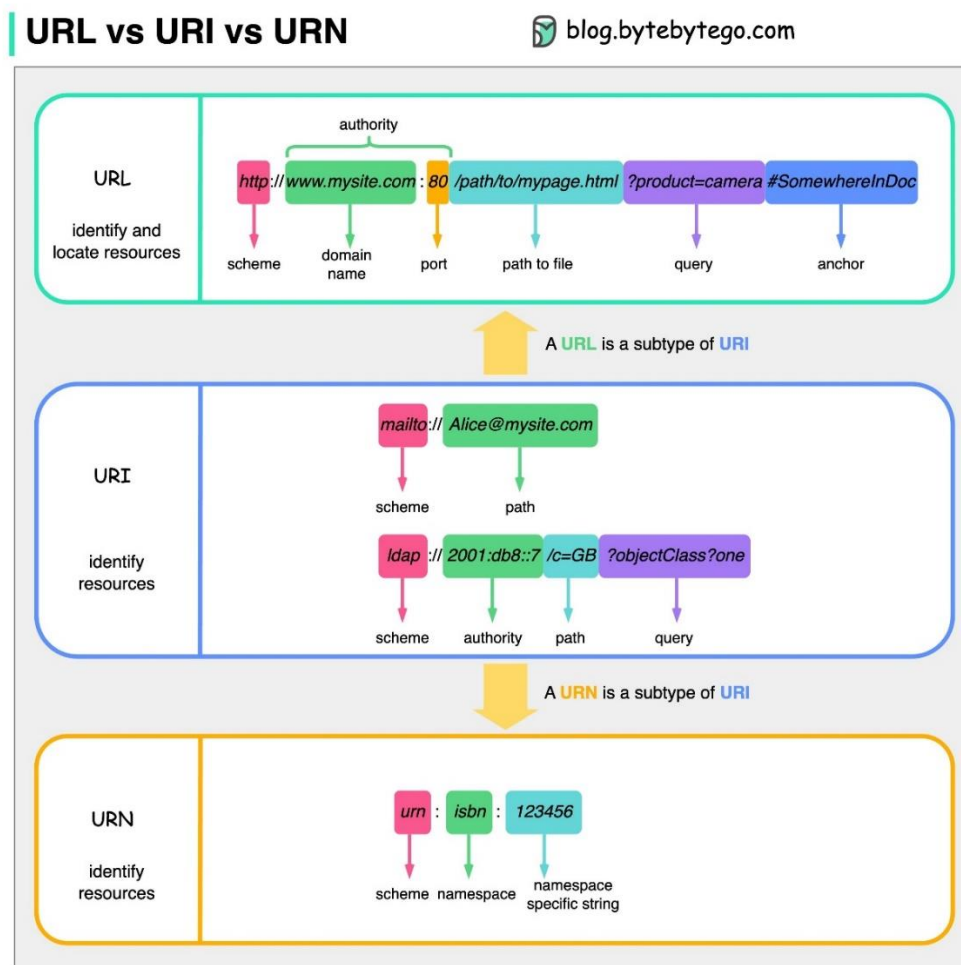
- URL: Una URL es una cadena de caracteres que se utiliza para identificar de manera única un recurso en Internet. Es decir, la localización de dicho recurso.
- URN: Los URN se utilizan para identificar recursos de manera abstracta, sin necesariamente proporcionar información sobre cómo acceder a ellos. Un ejemplo de URN es el ISBN

Enlaces:

[URI/URL/URN en Wikipedia.](#)

[URI/URL/URN en Geekflare.](#)

Modelos/imágenes:



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

Respuesta:

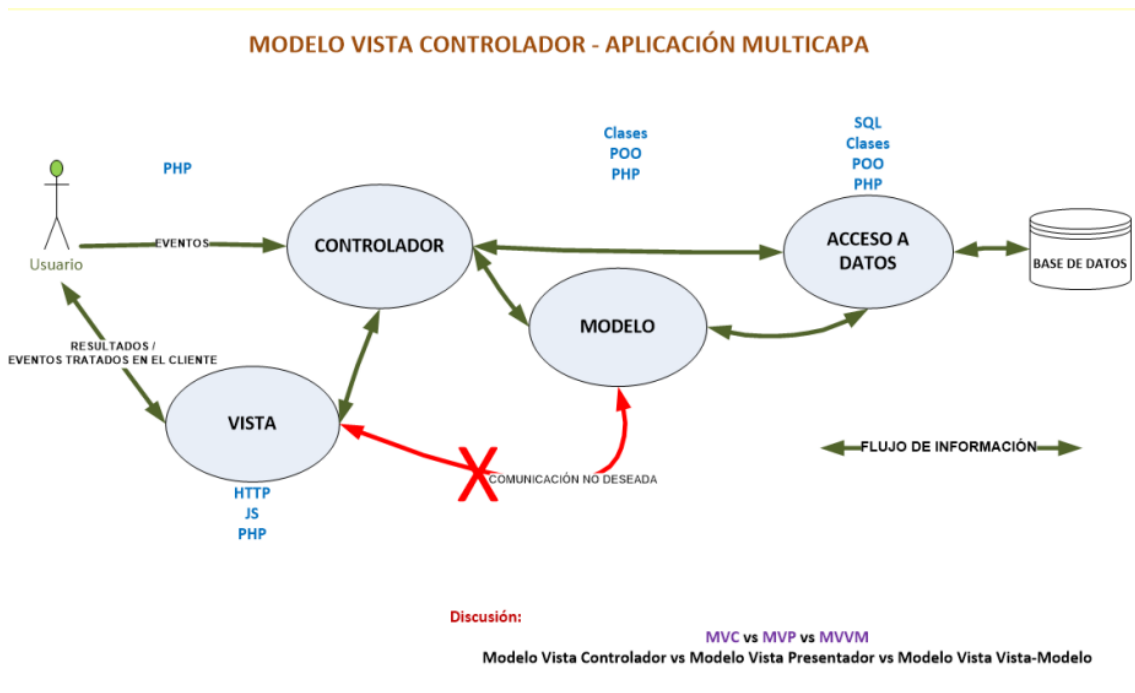
Un modelo de desarrollo de aplicaciones multicapa se utiliza para organizar y estructurar una aplicación en distintas capas. Pueden ser 4 capas como la aplicación que se muestra en la imagen o 3 como la arquitectura MVC.

- Vista: es lo que se muestra al usuario. Es el único archivo que contiene html y usa los datos del controlador.
- Controlador: es la que recibe los eventos y es donde tomamos las decisiones. Tiene que usar el modelo y la capa de acceso a datos. Es el que se ocupa de pasar los datos a la vista.
- Modelo: es donde se encuentran todas las clases de la aplicación.
- Acceso a datos: mueve la información de la persistencia o realiza queries. Utiliza el modelo y es utilizado por controlador.

Enlaces:

[Arquitectura multicapa en Wikipedia.](#)

Modelos/imágenes:



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

Respuesta:

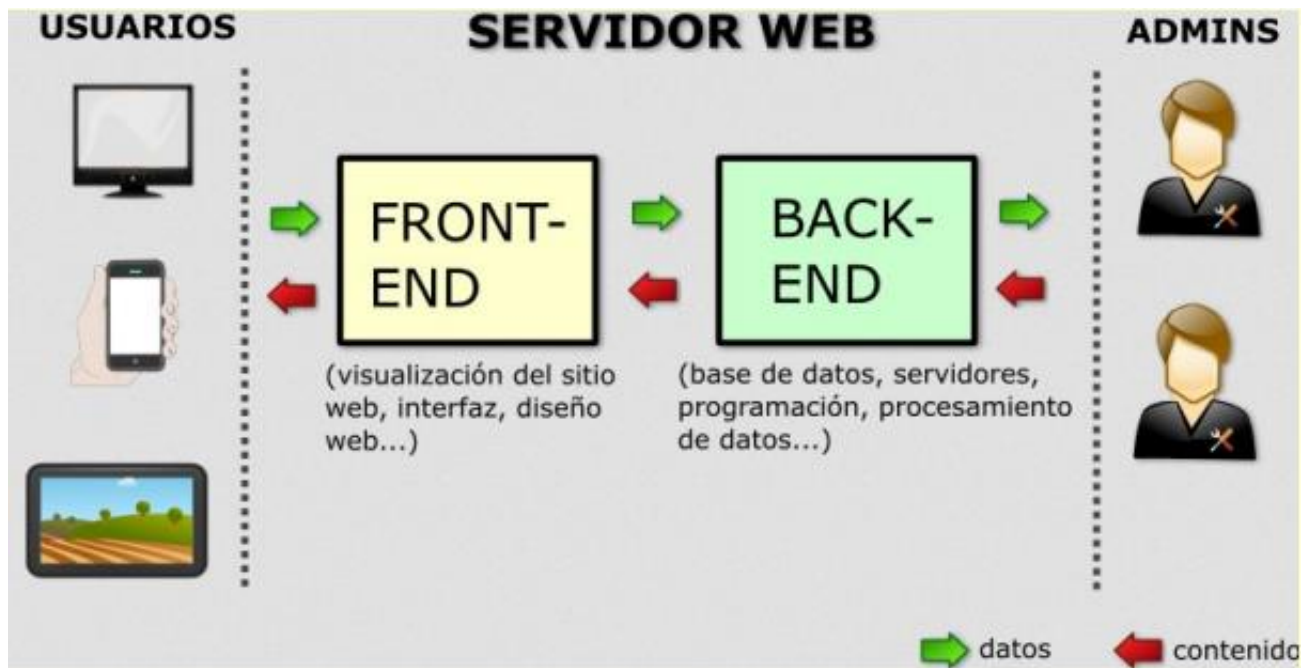
El modelo de división funcional front-end/back-end para aplicaciones web se refiere a la separación de las responsabilidades en el desarrollo de una aplicación web en dos partes principales:

- Front-end: Esta parte se encarga de la interfaz de usuario y la experiencia del usuario. Incluye el diseño, la presentación y la interacción visual de la aplicación.
- Back-end: Los desarrolladores crean la infraestructura que permite que la aplicación funcione, se comunica con bases de datos, maneja la autenticación de usuarios y procesa solicitudes del front-end.

Enlaces:

[Front-end y back-end en Wikipedia.](#)

Modelos/imágenes:



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

Respuesta:

- Página web estática: Es una página web que se entrega al navegador del usuario exactamente como está almacenada, es fija y estática. No tienen persistencia de datos. Siempre tiene la misma URL.
- Página web dinámica: Una página web dinámica muestra contenido que puede cambiar. Diferentes urls y tiene persistencia a datos.
- Aplicación web: Una aplicación web es una aplicación informática que se ejecuta en un navegador web. Tiene control de acceso, y es más compleja que una página web dinámica.
- Mashup: Son aplicaciones que aparte de ser multicapa integran tecnologías diferentes a la vez como una parte java otra php etc.

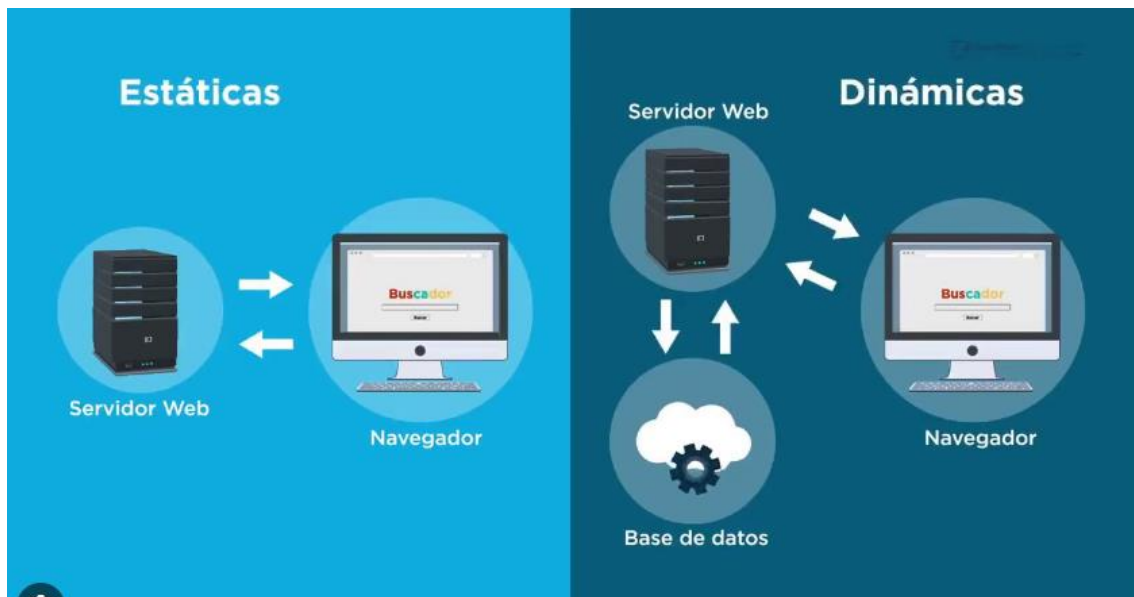
Enlaces:

[Página web estáticas vs dinámicas.](#)

[Significado de aplicación web.](#)

[Mashup en Wikipedia.](#)

Modelos/imágenes:



8. Componentes de una aplicación web.

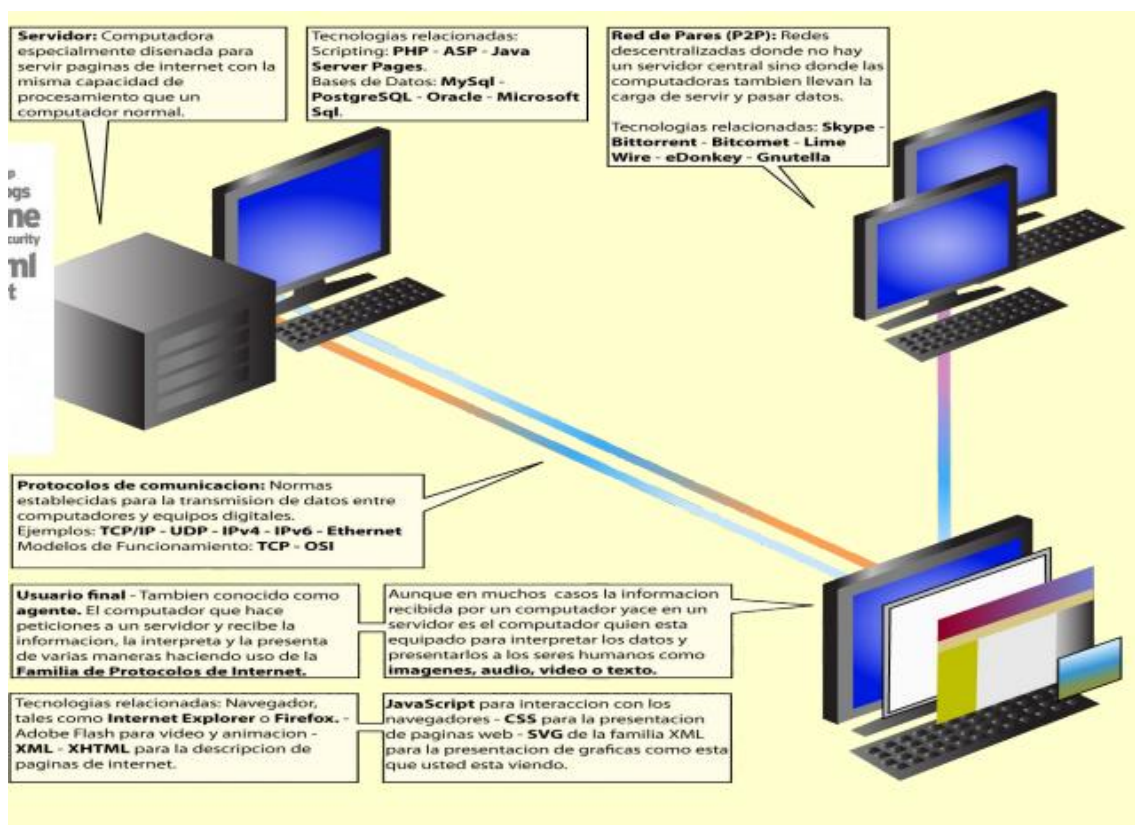
Respuesta:

- Sistema gestor de base de datos y base de datos: Es para almacenar todos los datos gestionarlos y realizar queries.
- Sistema de ficheros: Donde se encuentran todos los ficheros de la aplicación como ficheros PHP, HTML, CSS etc.
- Cliente web: un navegador para probar nuestra aplicación web.
- Servidor web: necesitamos un servidor web para gestionar y desarrollar nuestra aplicación web.
- Interprete de código: interpretar el código de nuestra aplicación esto tenemos que descargarnos la librería correspondiente de nuestro servidor web en mi caso la librería de php en Apache HTTP.

Enlaces:

[Componentes de una aplicación web según IBM.](#)

Modelos/imágenes:



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

Respuesta corta:

LADO DEL SERVIDOR:

- Java
- Ruby
- .NET
- Perl
- PHP
- Python
- JavaScript

LADO DEL CLIENTE:

- JavaScript
- CSS
- HTML
- ANGULAR
- VUE
- JQUERY
- REACT.

Enlaces:

[Lenguajes del lado del cliente y del lado del servidor.](#)

Modelos/imágenes:



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

Respuesta:

- Los lenguajes más usados son PHP, Java, .NET, Ruby y JavaScript
- PHP esta especialmente diseñado para el desarrollo web es el más usado para esto y es fácil de usar en la actualidad se suelen usar frameworks como Laravel o Symfony
- .NET es un framework propiedad de Microsoft en el que se usan lenguajes muy populares como C# y VisualBasic.
- Ruby es un lenguaje orientado a objetos fácil de leer fue muy popular en la década pasada ahora ya ha perdido popularidad se usa el framework Ruby on rails
- Java es un lenguaje de programación orientado a objetos que ofrece alta escalabilidad y rendimiento. Es muy utilizado en aplicaciones empresariales y grandes sistemas se suele usar con el frameworks Spring.
- JavaScript es versátil y se utiliza tanto en el lado del cliente como en el lado del servidor. Node.js es un entorno de ejecución de JavaScript en el lado del servidor que permite desarrollar aplicaciones webs rápidas no tan usado para crear grandes aplicaciones.

Enlaces:

[Lenguajes y características del lado servidor.](#)

Modelos/imágenes:



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

Respuesta:

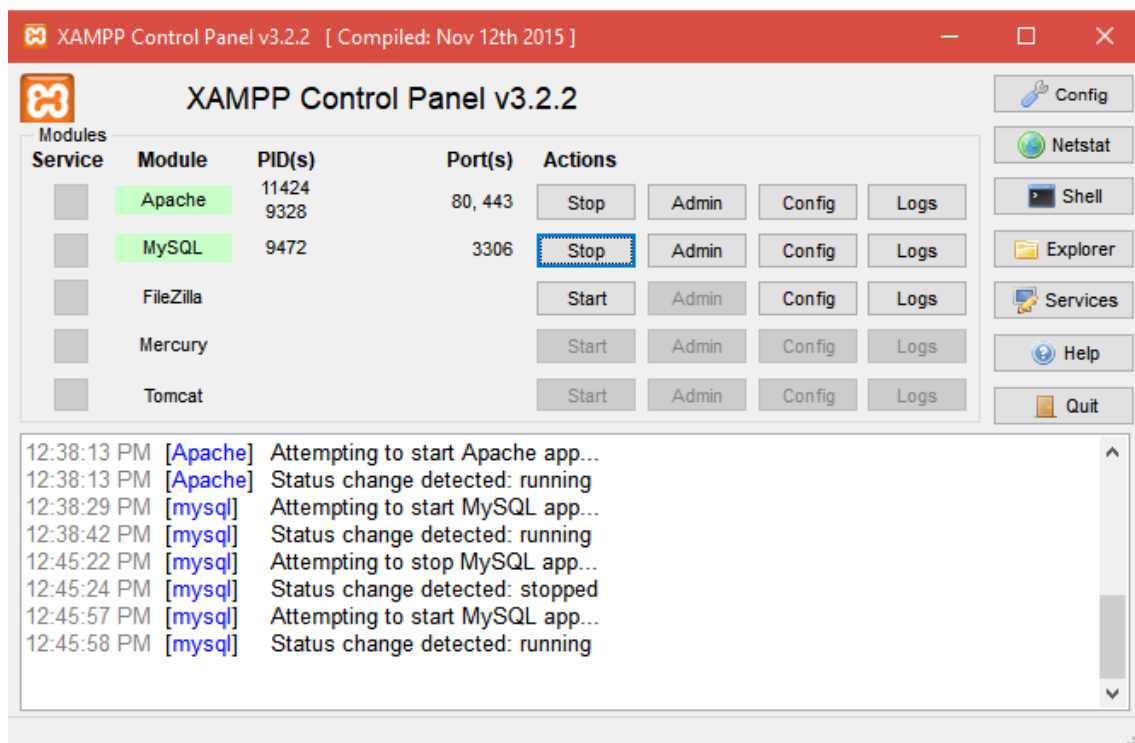
Es una plataforma versátil y fácil de usar que proporciona un entorno de desarrollo web completo para la creación y prueba de aplicaciones web. Es especialmente útil para desarrolladores que desean trabajar en sus proyectos de manera local antes de llevarlos a un servidor en producción. Características:

- Multiplataforma compatible con Windows, Linux y macOS.
- Fácil instalación y gran comunidad que ofrecen soporte y tutoriales.
- Compatibilidad con Apache uno de los servidores web más importantes, gestión de base de datos con MySQL o MariaDB conectable con phpMyAdmin para facilitar la gestión de bases de datos además de los lenguajes de programación más populares como Php.
- Seguridad XAMPP ha sido creado para un entorno local puede tener graves problemas de seguridad en un entorno de producción

Enlaces:

[Que es XAMPP.](#)

Modelos/imágenes:



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.

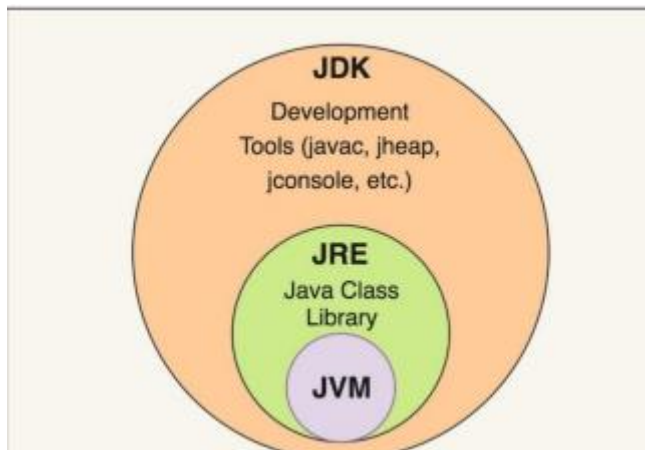
Respuesta:

En el entorno de desarrollo si deseas programar Java necesitas el JDK que es una versión de JVM, pero con añadidos para el desarrollo y en el entorno de explotación con el JVM que es el encargado de compilar es suficiente ya que solo necesitas ejecutar programas Java.

Enlaces:

[Diferencias entre JDK, JRE Y JVM.](#)

Modelos/imágenes:



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

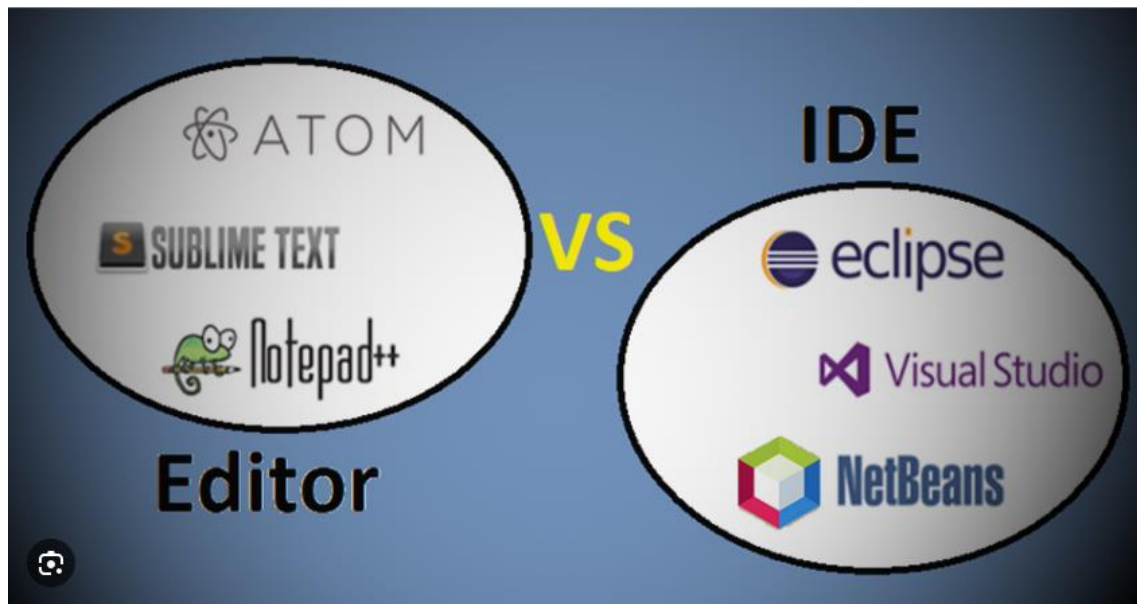
Respuesta:

- Visual Studio Code (VSCode): Este IDE de código abierto desarrollado por Microsoft se ha vuelto muy popular entre los desarrolladores debido a su extensibilidad y su amplio soporte para una variedad de lenguajes de programación. Su integración con Git y otras herramientas de desarrollo lo convierten en una opción atractiva para muchos.
- IntelliJ IDEA: Desarrollado por JetBrains, IntelliJ IDEA es conocido por su sólido soporte para Java y su capacidad para trabajar con una variedad de otros lenguajes de programación, como Kotlin, Groovy y Scala. También tiene versiones específicas para desarrolladores que trabajan con Android (Android Studio) y con el lenguaje de programación Kotlin.
- Eclipse: Este IDE de código abierto es ampliamente utilizado para el desarrollo de software en una variedad de lenguajes de programación, con un enfoque particular en Java. Ofrece una gran cantidad de complementos y es altamente personalizable.
- NetBeans: Otro IDE de código abierto, NetBeans, es conocido por su soporte para una amplia gama de lenguajes de programación, incluyendo Java, PHP y C/C++. Se destaca por su facilidad de uso y su capacidad para facilitar el desarrollo de aplicaciones web.
- Xcode: Este IDE es exclusivo para el desarrollo de aplicaciones para los sistemas operativos de Apple, como iOS, macOS, watchOS y tvOS. Es conocido por su integración completa con los diferentes servicios y frameworks de Apple, lo que lo convierte en una herramienta indispensable para los desarrolladores de aplicaciones de Apple.

Enlaces:

[Los IDEs según Wikipedia](#)

Modelos/imágenes:



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

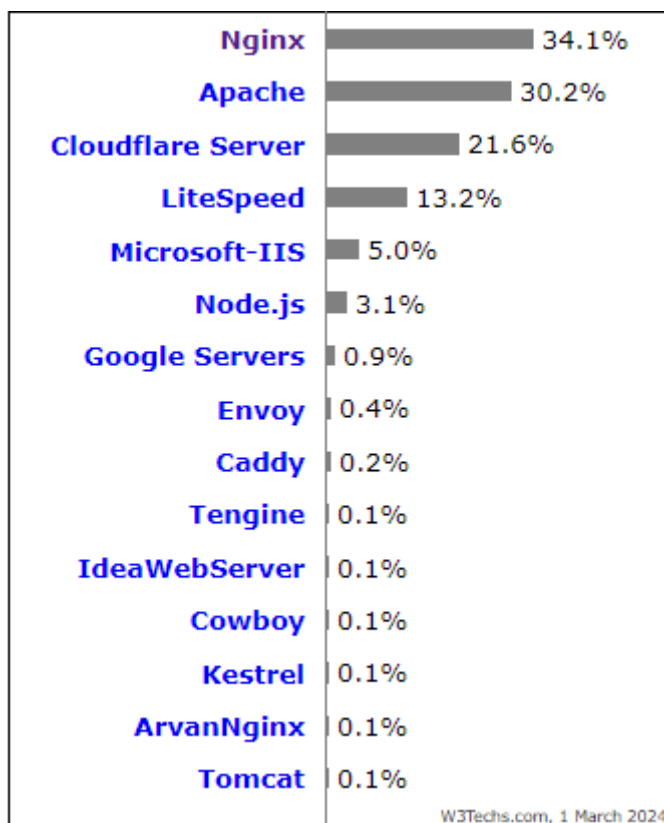
Respuesta corta:

- Nginx:
 - Nginx es un servidor web que se utiliza principalmente para los lenguajes PHP y Ruby, pero admite más.
 - Es software libre pero también existe una versión comercial llamada Nginx Plus.
 - Es un servidor web multiplataforma por eso es tan usado lo que quiere decir que interpreta muchos lenguajes.
- Apache HTTP Server:
 - En este servidor web principalmente se utilizan los lenguajes PHP y Ruby.
 - Es software libre por lo tanto no tienes que pagar para usarlo.

Enlaces:

[Servidores HTTP/HTTPS más utilizados.](#)

Modelos/imágenes:



15. Apache HTTP vs Apache Tomcat.

Respuesta corta:

Apache HTTP Server se utiliza principalmente para servir contenido estático y dinámico en una amplia gama de lenguajes, mientras que Apache Tomcat está especializado en el despliegue de aplicaciones web Java, incluidas servlets y tecnologías relacionadas. Sin embargo, a menudo se utilizan juntos, donde Apache HTTP Server actúa como un servidor frontal y Apache Tomcat se utiliza para gestionar la lógica del lado del servidor en aplicaciones Java.

Enlaces:

[Comparación Apache Tomcat y Apache HTTP.](#)

Modelos/imágenes:



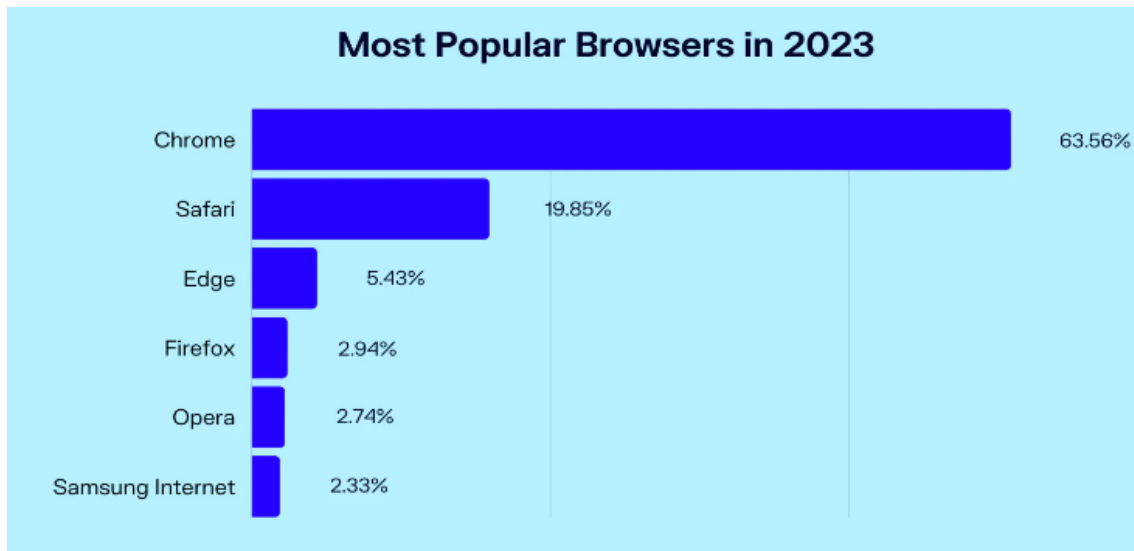
16. Navegadores HTTP /HTTPS más utilizados (características y grado de implantación actual).

Respuesta corta:

- Google Chrome:
Características: Rápido, seguro, soporte para extensiones, integración con la cuenta de Google, actualizaciones automáticas, amplia compatibilidad con estándares web.
Grado de Implantación: Ampliamente utilizado y líder del mercado en enero de 2022, con una participación de mercado superior al 60%.
- Safari:
Características: Exclusivo para dispositivos Apple, rendimiento rápido, énfasis en la privacidad, integración con el ecosistema de Apple.
Grado de Implantación: Ampliamente utilizado en dispositivos Apple, tenía una cuota de mercado de alrededor del 18-20% .
- Microsoft Edge:
Características: Integrado en Windows 10 y posteriores, rendimiento mejorado, soporte para extensiones, modo Internet Explorer para la compatibilidad con sitios heredados.
Grado de Implantación: Su cuota de mercado estaba en crecimiento y se situaba alrededor del 4-6%.
- Mozilla Firefox:
Características: Personalizable, código abierto, énfasis en la privacidad, soporte para extensiones, amplia compatibilidad con estándares web.
Grado de Implantación: Aunque su cuota de mercado había disminuido en comparación con Chrome, Firefox seguía siendo una opción popular con una participación de mercado de alrededor del 2-4%
- Opera:
Características: Rápido, seguro, características de ahorro de datos, VPN incorporada, bloqueo de anuncios integrado, soporte para extensiones.
Grado de Implantación: Opera tenía una cuota de mercado de alrededor del 2%.

Enlaces:

[Navegadores más utilizados.](#)

Modelos/imágenes:

17. Generadores de documentación HTML (PHPDoc): PHPDocumentor, ApiGen, ...

Respuesta corta:

Son herramientas que facilitan la generación de documentación en HTML a partir de comentarios en el código fuente en PHP, como PHPDocumentor y ApiGen."

- PHPDocumentor: Es una herramienta ampliamente utilizada que analiza el código fuente de PHP y genera documentación en formato HTML basada en los comentarios del código. Proporciona una amplia gama de etiquetas para documentar clases, métodos, propiedades y más, creando una documentación estructurada y legible.
- ApiGen: Similar a PHPDocumentor, ApiGen también crea documentación en HTML a partir de comentarios en el código fuente PHP. Ofrece flexibilidad en la personalización de plantillas y estilos de documentación, permitiendo generar documentación fácilmente legible y visualmente atractiva.

Enlaces:

[PHPDoc en zonaphp.](#)

Modelos/imágenes:



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

Respuesta corta:

Los sistemas de control de versiones son herramientas vitales en el desarrollo de software para gestionar cambios en el código fuente.

GIT: Es distribuido, lo que significa que cada usuario tiene una copia local completa del historial de versiones. Esto facilita el trabajo offline y permite ramificaciones (branches) y fusiones (merges) eficientes.

CVS (Concurrent Versions System): Fue uno de los primeros sistemas de control de versiones ampliamente utilizados. Aunque ya no es tan popular, fue fundamental en el desarrollo de software durante un tiempo. Sin embargo, su estructura centralizada y limitaciones en ramificaciones lo han vuelto menos adecuado para proyectos complejos.

Subversion (SVN): Ofrece una alternativa centralizada a CVS, mejorando algunas de sus limitaciones. Proporciona un sistema más ordenado y una estructura de directorios más flexible.

Enlaces:

[Control de versiones Seidor.](#)

Modelos/imágenes:

