

# **PREGUNTAS - TEMA 1**

## **DESARROLLO WEB EN ENTORNO SERVIDOR**

**25/09/2023**

**Ismael Ferreras García**



Realiza un estudio sobre los siguientes conceptos:

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

Son los protocolos más usados en internet y necesarios para su funcionamiento ya que es como se comunican los clientes web con los servidores web:

IP (Protocolo de Internet): **IP es el protocolo fundamental que se utiliza en Internet y redes para enrutar los paquetes de datos desde una fuente hasta un destino.** Hay dos versiones principales de IP, IPv4 (la versión que se utiliza ahora mismo) e IPv6 (creado para remplazar a IPv4 ya que no tiene suficiente capacidad).

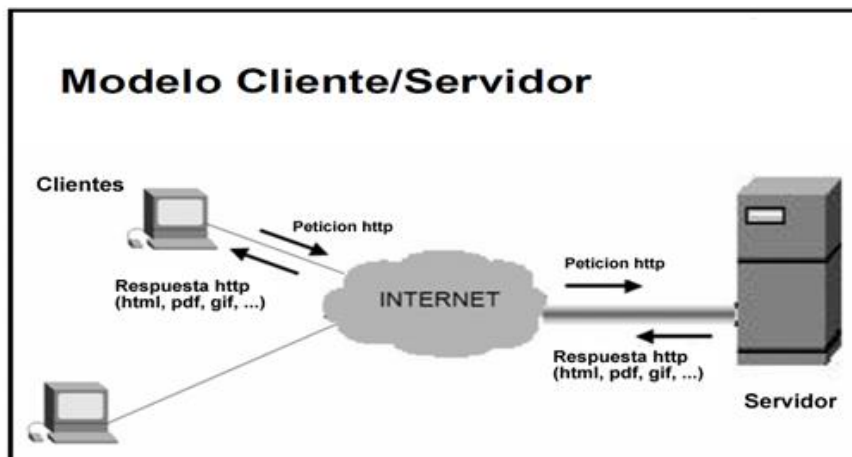
TCP (Protocolo de Control de Transmisión): **TCP es un protocolo de nivel de transporte que se encarga de establecer y mantener conexiones entre dos dispositivos en una red.** Garantizando que los datos se transmitan sin errores ni pérdidas.

HTTP (Protocolo de Transferencia de Hipertexto): **HTTP es un protocolo de nivel de aplicación que se utiliza para la transferencia de datos en internet.** Usa la estructura **cliente-servidor** entre un cliente (un navegador web) y un servidor web. HTTP es un protocolo sin estado, lo que significa que procesa cada solicitud independientemente de las anteriores. HTTP tiene 3 versiones principalmente HTTP/1, HTTP/2, HTTP/3 El servidor web utiliza el puerto 80 para la comunicación cliente-servidor

HTTPS (Protocolo de Transferencia de Hipertexto Seguro): **HTTPS es una versión segura de HTTP que utiliza cifrado SSL/TLS para proteger la privacidad y la integridad de los datos transmitidos entre el cliente y el servidor.** Actualmente es utilizado en todos los sitios web que utilizan información privada y en la mayoría de los que no. El servidor web utiliza el puerto 443 para la comunicación cliente-servidor

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

Es una estructura en la que el cliente (mediante un navegador o aplicación web) hace una petición http/https y el servidor web recibe y procesa la solicitud enviando la respuesta mediante http/https junto al contenido solicitado normalmente en formato HTML. El servidor actúa de forma pasiva nunca comienza la comunicación.



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

De esta lista get y post son los más utilizados

**HEAD:** Pide la respuesta

**GET:** Solicita un recurso/datos

**POST:** Envía datos al servidor para que sean procesados

**PUT:** Envía o actualiza datos al servidor para que sean procesados

**DELETE:** Borra el recurso

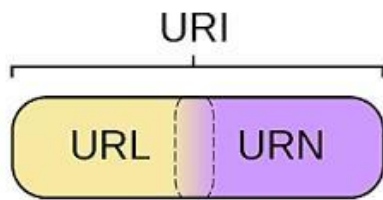
**TRACE:** Solicita al servidor que envíe de vuelta el mensaje (suele estar deshabilitado por seguridad)

**OPTIONS:** Devuelve las opciones de conexión HTTP que soporta la URL

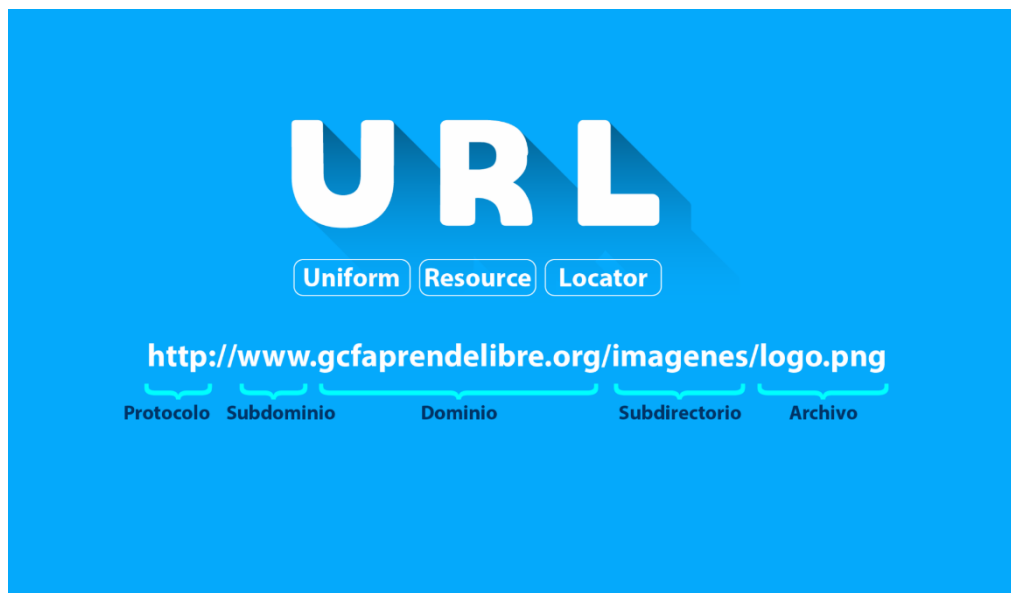
**CONNECT:** Establece una conexión de red con el recurso de destino

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

**URI es una cadena de caracteres utilizada para identificar un recurso de manera única** ya sea en la web como la URL de una página web o fuera de ella como el ISBN de un libro.



**Una URL es un tipo de URI para identificar un recurso en la red** proporcionando además la ubicación precisa de este y si está disponible



**Un URN es otro tipo de URI que se utiliza para identificar un recurso de manera única por su nombre, independientemente de su ubicación.** A diferencia de las URL, los URN no indican cómo encontrar el recurso o si está disponible, no es tan usado como la URL.

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

Formas de agrupar el contenido en capas

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

**Front-end** parte de la aplicación que usa un usuario normal

**Back-end** parte de la aplicación que usan los usuarios especiales administradores, servicio técnico...

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

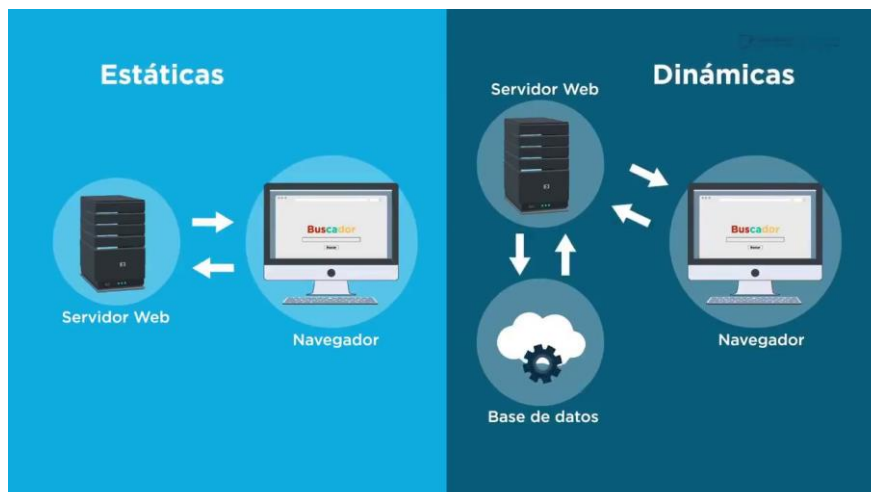
Estos términos se refieren a tipos de sitios web desde una página web estática siendo el más simple y que casi no permite al usuario interactuar hasta una aplicación web o un mashup siendo muy complejos, dinámicos e interactivos con el usuario.

**Página web estática:** Una página web estática es un sitio web que muestra contenido fijo y no cambia automáticamente según las interacciones del usuario. Por lo general, estas páginas se crean con HTML y CSS, y su contenido se mantiene sin cambios hasta que el propietario del sitio web lo actualiza manualmente. Suelen ser utilizadas para sitios web simples, como páginas de información o portafolios en línea.

**Página web dinámica:** Una página web dinámica es un sitio web que muestra contenido que puede cambiar en función de las acciones del usuario o la información de una base de datos. Se utilizan tecnologías como PHP, ASP.NET, Ruby on Rails o JavaScript para crear páginas web dinámicas. Estas páginas son capaces de mostrar contenido diferente a diferentes usuarios o en diferentes momentos, como una plataforma de redes sociales o una tienda en línea.

**Aplicación web:** Una aplicación web es una aplicación interactiva que se ejecuta en un navegador web. A menudo, ofrece funcionalidades similares a las aplicaciones de software tradicionales, pero no requiere una instalación en el dispositivo del usuario. Las aplicaciones web pueden ser tanto estáticas como dinámicas y pueden proporcionar una amplia gama de funciones, desde procesamiento de datos hasta interacción en tiempo real.

**Mashup:** Un mashup es una aplicación o sitio web que combina datos o funcionalidades de varias fuentes diferentes para crear una experiencia integrada. Los mashups suelen tomar información de diversas fuentes web, como APIs (interfaces de programación de aplicaciones), y la presentan de una manera combinada o enriquecida. Por ejemplo, un mashup de mapas podría combinar datos de mapas de Google con datos de ubicación de restaurantes de Yelp para mostrar restaurantes en un mapa.



## 8. Componentes de una aplicación web.

Cliente web que es normalmente un navegador su función es mandar peticiones al servidor y interpretar ciertos lenguajes como HTML, CSS y JS

Servidor web: Para recibir las peticiones de los clientes web (navegadores normalmente) y procesarlas

Base de datos: Almacena los datos de la aplicación.

Ficheros en un lenguaje de programación en el que está escrita la aplicación y modulo encargado de ejecutar el código esto se encuentra integrado en el servidor.



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

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

#### Lado del cliente

El lenguaje más utilizado es JavaScript ocupando todo el mercado con más de un 99% de presencia

#### Lado del servidor

Los lenguajes más usados son PHP, Java, ASP.NET, Ruby y JavaScript

PHP esta especialmente diseñado para el desarrollo web es el mas usado para esto y es fácil de usar en la actualidad se suelen usar frameworks como Laravel o Symfony

ASP.NET es un framework en el que se usan lenguajes muy populares como C# y .net

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.

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 web altamente escalables y rápidas.

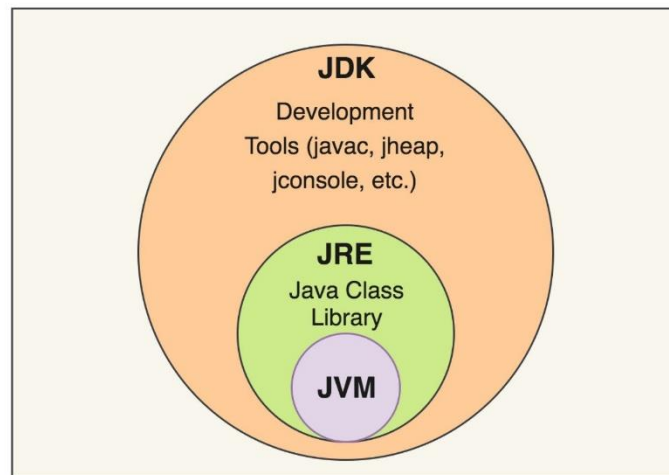
© W3Techs.com	usage	change since 1 August 2023
1. <a href="#">PHP</a>	76.9%	-0.5%
2. <a href="#">ASP.NET</a>	6.8%	-0.1%
3. <a href="#">Ruby</a>	5.5%	+0.1%
4. <a href="#">Java</a>	4.7%	
5. <a href="#">JavaScript</a>	3.0%	+0.3%
percentages of sites		



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

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.

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 es suficiente ya que solo necesitas ejecutar programas Java



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

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

15. Apache HTTP vs Apache Tomcat