

# Índice

TEMA 1. INTRODUCCIÓN.....	3
1. Internet y la web .....	3
1.1 ¿Qué es un protocolo de comunicaciones?.....	3
1.2 ¿Qué es el internet? .....	4
1.3 ¿Qué es la web? .....	5
1.4 Términos a conocer .....	5
1.5 Estándares web .....	5
2. Modelo cliente – servidor .....	6
3. Desarrollo web.....	7
3.1. Protocolo HTTP. Estructura. ....	7
3.2. Métodos de HTTP.....	8
3.3. HTTP/2 .....	8
4. Arquitecturas web.....	8
4.1. Applets.....	8
4.2. Servlet.....	9
4.3. JavaServer Pages.....	9
4.4. Servicios web .....	9
4.5. REST .....	9
TEMA 2. DISEÑO Y DESARROLLO DE PÁGINAS WEB. ....	10
5. Diseño de páginas web.....	10
6. Desarrollo de documentos HTML .....	11
6.1. Marcado semántico .....	11
7. Cascading Style Sheets (CSS) .....	12
7.1. Content Management Systems (CSM).....	13
8. Cloud computing .....	13
8.1. Arquitectura de los servicios. Tipos de infraestructuras cloud. ....	13

1. IaaS (Infraestructura como servicio).....	14
2. PaaS (Plataforma como servicio).....	14
3. SaaS (Software como servicio) .....	14
8.2. Ventajas y desventajas del Cloud Computing .....	15
8.3. Aplicaciones .....	15
TEMA 3. PROGRAMACIÓN WEB EN EL SERVIDOR.....	16
9. Aplicaciones y sitios web.....	16
9.1. Terminología actual .....	16

# TEMA 1. INTRODUCCIÓN

## 1. Internet y la web

Una red de ordenadores es un conjunto de ordenadores conectados entre sí para el transporte de datos con el objeto de compartir información, recursos y ofrecer servicios.

Cuando un pc se conecta a internet, lo hace mediante un modelo. Nosotros estudiamos el modelo OSI, que tiene las siguientes capas:

- 1- Aplicación
- 2- Presentación
- 3- Sesión
- 4- Transporte
- 5- Red
- 6- Enlace
- 7- Físico

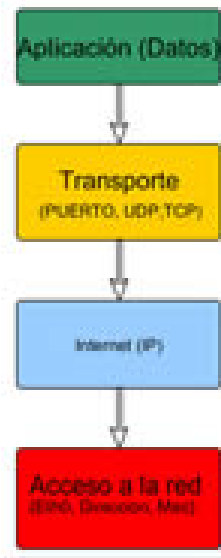
### *1.1 ¿Qué es un protocolo de comunicaciones?*

Una **red de ordenadores** es un conjunto de ordenadores conectados entre sí para el transporte de datos con el objeto de compartir información, recursos y ofrecer servicios, que lo hacen mediante **conjunto de reglas(protocolo de comunicaciones)** que están obligadas a cumplir todas las máquinas y programas que se comunican con otras máquinas. Se imponen para que la comunicación no sea caótica y por tanto imposible. Los protocolos de comunicación se pueden dividir en varias categorías, entre la que destaca el **modelo OSI**.

## Las 7 capas del modelo OSI



y el protocolo **TCP/IP**, que está compuesto por 4 capas:



## 1.2 ¿Qué es el internet?

Es un conjunto descentralizado de redes de comunicación interconectadas entre sí que usan los protocolos TCP/IP. Es un concepto previo a la **web**.

Es una red descentralizada, con 0 nodos.

## 1.3 ¿Qué es la web?

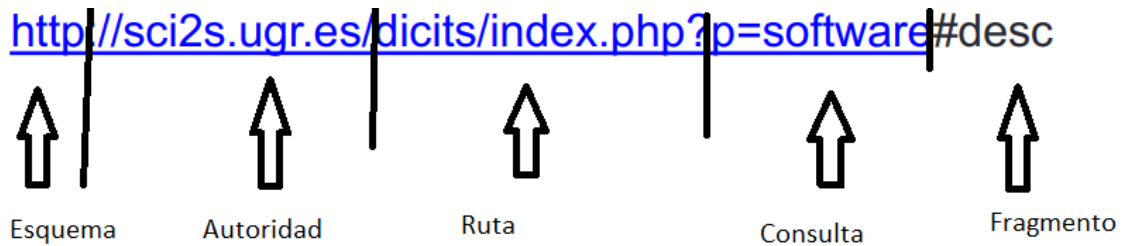
Es un sistema de distribución de información basada en **hipertexto** accesible a través de internet, creada por Tim Berners –Lee en 1980.

## 1.4 Términos a conocer

- Página web: es un documento web en HTML
- Lenguajes de marcado: es una forma de codificar un documento que, junto con el texto, incorpora etiquetas que contiene información adicional sobre la estructura del texto o presentación.
- Servidor web
- Navegador: es una aplicación para obtener y mostrar información de la web (Chrome). Debe visualizar los documentos obtenidos, interpretando su estructura y mostrando los recursos que integra.
- Hiperenlace: es el elemento de un documento electrónico que hace referencia a otro recurso.
- URL: identificador de recursos uniforme. Son direcciones que apuntan a recursos variables en el tiempo.
- Buscador web: es un programa que busca archivos alojados en servidores (Google).
- Servicio web: es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones.

## 1.5 Estándares web

- URI: sistema universal para hacer referencia a recursos en la web. Los tipos son el URL (indican exactamente dónde se encuentra el recurso. Dirección del recurso) y la URN(no lo indica). Las partes del URI son:



- HTTP: protocolo de comunicación **sin estado** (no se mantiene información entre solicitudes) entre navegador y servidor, basado en TCP/IP para el acceso a contenidos a través de la web. Se trata de un protocolo **sin conexiones**.
- HTML: Lenguaje de marcado de hipertexto para definir estructura, aspectos de apariencia. Puede incluir contenido dinámico a través de scripts y contenido tipo MIME.

Los elementos de un documento HTML son los siguientes:

1. Cabeceras: contiene información que describe el documento HTML (títulos, hojas de estilo, guiones, otra metainformación)
  2. Cuerpo: contiene toda la información del documento.
  3. Formularios
  4. Scripts
- XHTML: Es una combinación entre el HTML y XML. Tiene unas indicaciones más estrictas que el HTML.
  - XML: permite definir la gramática de lenguajes para estructurar documentos.

## 2. Modelo cliente – servidor

Es un modelo de arquitectura web donde las tareas se reparten entre los servidores y los clientes. Un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta.

El servidor más usado a día de hoy es Apache, que es un servidor de código abierto multiplataforma, altamente configurable y modular.

### 3. Desarrollo web

Es un término extenso que engloba las tecnologías y procesos de creación de páginas web y aplicaciones web.

El desarrollo web incluye varios aspectos:

- Funcionales
- Estéticos
- Técnicos
- Diseño: se trata de cuidar la interacción con el usuario (colores, compatibilidades, etc). Para hacer la navegación más cercana y cómoda.
- Desarrollo en servidor: Php, python, ruby, java, etc.
- Desarrollo en cliente: JS, HTML5, CSS, etc.
- Entorno de trabajo para el desarrollo: Django (python), Zend (php), etc.
- Bases de datos: Mysql, MariaDB, etc.
- Interfaz de usuario

#### 3.1. Protocolo HTTP. Estructura.

Los mensajes se representan como texto. Las peticiones tienen la siguiente estructura:

➤ Línea inicial finalizada con CRLF



- Varias líneas de cabecera: dan información sobre la solicitud o respuesta. Si hay espacios al principio significa que es la continuación.

User-agent: Mozilla/3.0Gold

- Línea en blanco
- Cuerpo del mensaje

## 3.2. Métodos de HTTP

- **GET:** recuperar información del lugar identificado con la URI. Se usa para enviar formularios.
- **HEAD:** igual que GET pero solo pide las cabeceras.
- **POST:** Enviar datos al servidor.

## 3.3. HTTP/2

Tiene como objetivos crear mecanismos de negociación, reducir la latencia de descarga de páginas y mantener la compatibilidad con HTTP/1.

# 4. Arquitecturas web

Diseñar una web es una disciplina cuyo objeto es la planificación y diseño de los sitios y aplicaciones web. Incluye los aspectos técnicos, estéticos y funcionales.

## 4.1. Applets

Es un componente de una aplicación que se ejecuta en el contexto de otro programa (por ejemplo, un navegador). No se puede ejecutar de forma independiente y realiza una tarea muy específica.



## 4.2. Servlet

Son applets que corren en el servidor. Se alojan en contenedores de servlets.

## 4.3. JavaServer Pages

Es una tecnología para el desarrollo de software que crean contenido dinámico para la web. Basado en HTML y XML.

## 4.4. Servicios web

Es un método de comunicación entre dispositivos en la web. Están permanentemente en ejecución.

## 4.5. REST

Es un estilo de arquitectura de software para sistemas distribuidos. Es el modelo de referencia para el diseño de servicios web. **Protocolo de transferencia de información entre aplicaciones web.**

### Ventajas

- Separación entre cliente y servidor
- Visibilidad, fiabilidad y escalabilidad (capacidad de atender demandas y liberar recursos)

# TEMA 2. DISEÑO Y DESARROLLO DE PÁGINAS WEB.

## 5. Diseño de páginas web

¿Qué es una página web? Es un documento diseñado para su difusión por la red, accediendo mediante navegadores. Se escribe en HTML o XHTML. Se diferencian en **estáticas** (prefijado y almacenado en disco) y **dinámicas**, que genera el contenido poco a poco.

¿Qué es un sitio web? Es un conjunto de páginas web relacionadas entre sí. Existen multitud de tipos de sitios web y tienen una estructura jerárquica.

Para el diseño de una página web debemos tener en cuenta las siguientes capas:

1. Capa estructural: es donde decidimos de qué trata nuestra web, así como la estructura del home y menú principal.
2. Capa de comportamiento: aquí se define cómo se comportará la web. Se inicia el proceso de dinamismo de la web.
3. Capa de presentación: representa el estado final (la interfaz) del diseño que se forma por la capa estructural y capa de comportamiento.

Para diseñar una web, debemos tener en cuenta ciertos factores importantes:

1. Estándares aplicables, para facilitar el trabajo a navegadores
2. Resoluciones de pantalla
3. Responsividad: buena adaptación a todos tamaños de pantallas.
4. Accesibilidad
5. Internacionalización

En general, todo lo que no sea a más bajo nivel.

## Recomendaciones generales

6. Diseño de un plan: antes de iniciar a escribir, piensa un plan
7. Estudiar la audiencia
8. Hacerlo de tal forma para no hacer pensar al usuario
9. Hacer una navegación fácil para el usuario
10. Poner enlaces directos para ir hacia adelante y atrás
11. Tener en cuenta el funcionamiento en móviles

## 6. Desarrollo de documentos HTML

La definición de HTML la hemos dado antes.

Actualmente, la versión usada es la 5.2 del HTML5. Esta versión incluye:

- Elemento **CANVAS** para dibujos en 2D
- Elementos **video y audio**
- Soporta almacenamiento local
- Elementos específicos de contenido como **article, footer, header, nav y section**
- Controles de formularios
- Soporte de **SVG**
- Nuevas APIs

### 6.1. Marcado semántico

Es el uso de etiquetas HTML para reforzar la semántica o el significado. Se trata de escribir correctamente las etiquetas para ayudar al navegador y al ser humano a identificar el contenido del documento. Como hemos indicado antes, este marcado semántico se ha ampliado en el HTML5 y las etiquetas footer, aside, etc.

## 7. Cascading Style Sheets (CSS)

Es un lenguaje de estilo que define la presentación de documentos HTML.

Obtenemos un control más preciso que el soportado por el HTML. Su objetivo es independizar estructura del contenido.

Se pueden incluir en:

- En línea(atributo "style")

```
<body style="background-color: #F00000;">
```

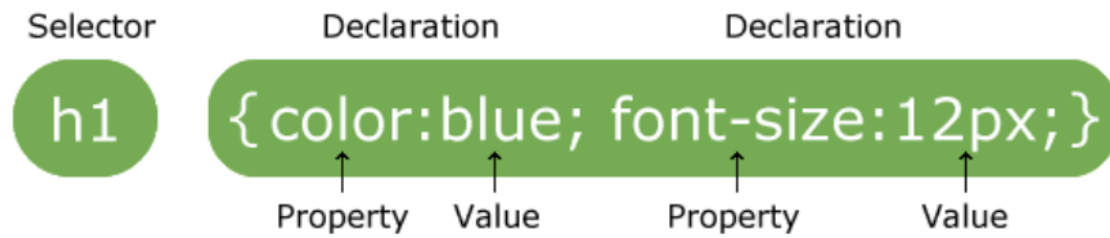
- Interno(elemento "style")

```
<head>
  <title>Ejemplo</title>
  <style type="text/css">
    body {background-color: #F00000;}
  </style>
</head>
```

- Externo(enlace a hojas de estilo)

```
<html>
  <head>
    <title>Mi documento</title>
    <link rel="stylesheet" type="text/css"
      href="style/style.css" />
```

La estructura de una sentencia CSS es la siguiente:



## 7.1. Content Management Systems (CSM)

Son aplicaciones que facilitan el desarrollo de la creación y edición de sitios y páginas webs. Tienen **dos componentes principales**: aplicación de gestión y aplicación de despliegue. Joomla, drupal, Wordpress, etc. Son como CRM.

## 8. Cloud computing

Es un **modelo de prestación de servicios** de negocio que permite al usuario acceder a un catálogo de servicios, de forma flexible y adaptativa pagando únicamente por el consumo efectuado.

Aspectos claves del cloud computing:

- Servicios bajo demanda
- Elasticidad/escalabilidad: se adapta a la demanda cambiante. Cada cliente se denomina “tenant” y los recursos son virtualizados.
- Fácil accesibilidad

### 8.1. Arquitectura de los servicios. Tipos de infraestructuras cloud.



### 1. IaaS (Infraestructura como servicio)

Ofrece recursos informáticos equivalentes a hardware virtualizado. Una ventaja es su escalabilidad. Se puede ampliar la capacidad en función de las necesidades. El proveedor del servicio es el dueño del equipo.

### 2. PaaS (Plataforma como servicio)

Entorno software con herramientas de desarrollo para una plataforma dada. Habitualmente incluye: S.O., lenguaje de programación/entorno de ejecución, base de datos, servidor web.

Microsoft Azure  
Amazon Web Services  
Google App Engine  
Heroku  
OpenShift (Red Hat)  
Cloud Foundry  
SalesForce

### 3. SaaS (Software como servicio)

Los proveedores instalan y gestionan software de aplicaciones en la nube, accesible desde los clientes de la nube. Se paga por el uso.

Google Apps: Gmail, Google Docs, ...  
Dropbox, Google Drive  
Quickbooks online (Salesforce.com)  
Evernote

## 8.2. Ventajas y desventajas del Cloud Computing

Las principales ventajas son una **reducción en los costes** en:

- Infraestructuras
- Ordenadores de usuario
- Licencias de software
- Energía
- Personal de informática
- 

Una **mejorada gestión y unas mejores prestaciones**:

- Adaptación elástica
- Capacidad de almacenamiento mejorada
- Mayor seguridad en los datos
- Disponibilidad 24/7

Las principales desventajas son:

- Una conexión constante a internet
- No funciona bien con bandas de ancha pequeñas
- Ofrecen menos características que aplicaciones de escritorio
- Cuestiones de privacidad y seguridad

## 8.3. Aplicaciones

# TEMA 3. PROGRAMACIÓN WEB EN EL SERVIDOR

## 9. Aplicaciones y sitios web

Una aplicación web es un documento con capacidad interactiva. Los elementos de una aplicación web son:

- Interfaz: el propio navegador hace de interfaz.
- Servidor web: es el que pone en marcha la aplicación web. Todo el código permanece aquí y nunca sale.
- Base de datos
- Lenguaje de programación
- Despliegue

### 9.1. Terminología actual

Para el desarrollo web:

- Back-end developer: ámbito del servidor
- Front-end developer: ámbito del cliente
- Full-stack developer: ambos

Relacionadas:

- Devops: Tareas de desarrollador y operador de sistemas.
- Continuous-integration: Hacer integraciones (compilación y ejecución de pruebas) automáticas de un proyecto lo más frecuentemente posible para acelerar la detección de fallos.

### 9.2. Lenguaje PHP

Es un lenguaje para aplicaciones web en el servidor de propósito general. Son procesados PREVIAMENTE al HTML. Se localiza en el interior del documento HTML. Es un lenguaje INTERPRETADO (se necesita un intérprete para que se ejecute). El resultado final será HTML. Es un lenguaje no tipado. No hay acceso directo a dispositivos como en otros



lenguajes. Las entradas de son parámetros recibidos a través de peticiones HTML (variables), datos de un formulario o archivos locales.

Las salidas es un código HTML que se inserta en el documento como respuesta del servidor.

### 9.2.1. Elementos del lenguaje PHP

- Sentencias

Se inician -> <?php

Se terminan --> ?>

- Variables: lenguaje no tipado. No hay verificación de tipos.
- Operadores
- Arrays: al no ser tipado, se puede meter lo que sea dentro de un mismo array (como python). Esto se llama heterogéneo. El contenido es accesible mediante índices numéricos o contenedor asociativo.

```
$matriz = array(array(1, 2, 3),  
                array(4, 5, 6));
```

- Funciones
- Estructuración: tiene un diseño modular.

- `include` (aviso)
- `require` (Error fatal)
- `include_once` (garantiza la inclusión una sola vez)
- `require_once`

- Formularios: es el principal medio de interacción con PHP.