

# Tema 01. Arquitecturas y Herramientas de Programación Web

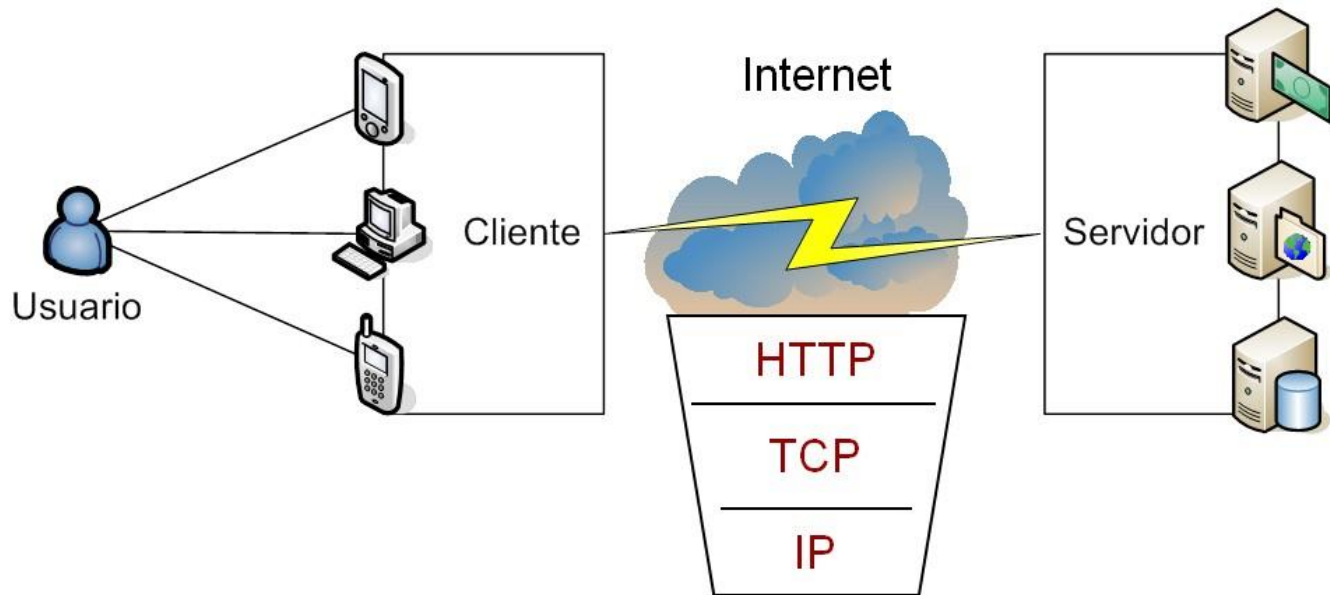
Desarrollo Web Entorno Servidor. 2º DAW 19/20

# Modelos de programación cliente-servidor

---

- ▶ **World Wide Web:**
  - ▶ Conjunto de recursos interconectados que conforman el conocimiento humano actual.
  - ▶ El funcionamiento de la web es debido a la coexistencia de componentes hardware y software:
    - ▶ *Componentes Físicos:* Hubs, repetidores, puentes, pasarelas, encaminadores o routers.
    - ▶ Protocolos de comunicaciones: TCP, IP, HTTP, FTP, SMTP.
    - ▶ Sistema de nombres de dominio (DNS).
- ▶ El desarrollo web debe tener en cuenta la distribución de estos elementos y la función que tienen cada uno de ellos.
- ▶ Configuración arquitectónica más habitual:
  - ▶ *Cliente/Servidor.*
    - ▶ *cliente* es un componente consumidor de servicios.
    - ▶ *servidor* es un proceso proveedor de servicios.
  - ▶ Relación cimentada mediante el intercambio de mensajes.

# Modelos de programación cliente-servidor



# Modelos de programación cliente-servidor

---

- ▶ Funcionalidades en los entornos cliente/servidor (*capas*):
  - ▶ **Capa de presentación:**
    - ▶ Es la que ve el usuario.
    - ▶ Suele estar situada en el cliente
    - ▶ La programación de esta capa se centra en el formateo de la información enviada por el servidor y 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.
    - ▶ Habitualmente es donde se reciben las peticiones del usuario y desde donde se envían las respuestas apropiadas tras el procesamiento de la información proporcionada por el cliente
    - ▶ Esta capa puede ser programada tanto en el entorno cliente como en el servidor.
  - ▶ **Capa de persistencia o de datos:**
    - ▶ Es la capa donde residen los datos y la encargada de acceder a los mismos.
    - ▶ Donde reside el sistema gestor de base de datos

# Modelos de programación cliente-servidor

---

- ▶ **Clasificación de los modelos de programación Web o Cliente/Servidor:**
  - ▶ Según el tamaño de los componentes:
    - ▶ Fat Client (thin Server), el mayor peso de la aplicación se ejecuta en el cliente.
    - ▶ Fat Server (thin client), funcionalidad asociada al cliente limitada a la presentación de la información enviada por el servidor.
  - ▶ Según la naturaleza del servicio ofrecido: en función de las capacidades ofrecidas por el servidor.
    - ▶ Servidores de ficheros, servidores de bases de datos, servidores de transacciones, servidores de objetos, servidores web...
  - ▶ Reparto de funciones entre cliente y servidor: gestionar y distribuir las responsabilidades de cada una de las prestaciones funcionales entre el cliente y el servidor.
    - ▶ Modelo en dos capas. Agrupa las capa de negociado y de datos
    - ▶ Modelo en tres capas.

# Páginas Web Estáticas

---

- ▶ Las páginas estáticas se encuentran almacenadas en su forma definitiva, tal y como se crearon, y su contenido no varía.
- ▶ Son útiles para mostrar una información concreta, y mostrarán esa misma información cada vez que se carguen.
- ▶ La única forma en que pueden cambiar es si un programador la modifica y actualiza su contenido.

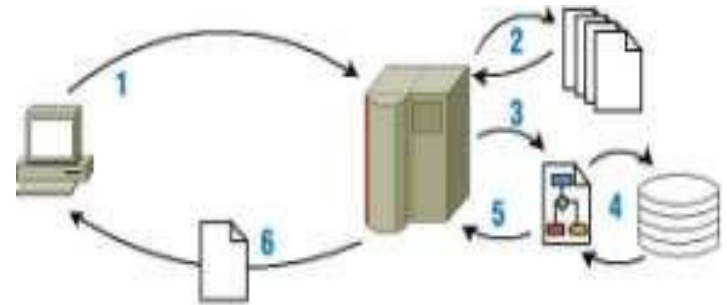
# Páginas Web Dinámicas

---

- ▶ Su contenido cambia en función de diversas variables:
  - ▶ El navegador que estás usando
  - ▶ El usuario con el que te has identificado
  - ▶ Las acciones que has efectuado con anterioridad.
- ▶ Tipos:
  - ▶ Con código de ejecución en cliente:
    - ▶ Incluyen código que se ejecuta en el navegador
    - ▶ Normalmente ese código es **javascript** que se incluye junto con el HTML.
    - ▶ La extensión de este tipo de páginas suele ser htm o html
  - ▶ Con código de ejecución en servidor:
    - ▶ El HTML de estas páginas se forma como resultado de la ejecución de un programa, y esa ejecución tiene lugar en el servidor web
    - ▶ Para la programación se pueden usar multiples lenguajes java, asp, php
    - ▶ Muchas de estas páginas tienen extensiones como .php, .asp, .jsp, .cgi o .aspx y no htm o html como las anteriores.

# Funcionamiento Web Dinámica

- ▶ El cliente web solicita a un servidor web una página.
- ▶ El servidor busca esa página y la recupera.
- ▶ En el caso de que se trate de una página web dinámica, es decir, que su contenido deba ejecutarse para obtener el HTML que se devolverá, el servidor web contacta con el módulo responsable de ejecutar el código y se lo envía.
- ▶ Como parte del proceso de ejecución, puede ser necesario obtener información de algún repositorio (Base de Datos).
- ▶ El resultado de la ejecución será una página en formato HTML, similar a cualquier otra página web no dinámica.
- ▶ El servidor web envía el resultado obtenido al navegador, que la procesa y muestra en pantalla.





# Lenguajes de programación de entorno servidor

---

- ▶ Lenguaje de programación correspondiente a un entorno servidor:
  - ▶ “Aquel cuyo código, bien sea como objeto precompilado o bien como código interpretado, es ejecutado por un software específico en el componente que actúa como servidor”.
- ▶ Tipos:
  - ▶ Lenguajes de scripting.
  - ▶ Aplicaciones CGI y derivados.
  - ▶ Aplicaciones híbridas de código repartido.

# Lenguajes de programación de entorno servidor

---

## ► Lenguajes de scripting:

- El código es interpretado y que se intercala con una plantilla de código HTML con la estructura básica de la página que se envía al servidor
- Algunos de estos lenguajes podrían ser:
  - PHP (PHP Hypertext Processor). De los más extendidos con gran comunidad desarrolladores. Software libre y código abierto. Soportado por la mayoría servidores actuales. Permite poo.
  - ASP (Active Server Pages). Tecnología propietaria de Microsoft. Sólo ejecutado en servidores IIS. Se trata del antiguo Visual Basic en su versión scripting.
  - Perl. Parecido al PHP y se suele utilizar para crear aplicaciones CGI
  - Python. Facilita la programación orientada a objetos. Lenguaje interpretado. Software libre y cada vez más popular.
  - JSP (Java Server Pages). Uso del lenguaje java como lenguaje scripting. Parecido a PHP y ASP pero con distinta forma de interpretación. Rendimiento mejorado.

# Lenguajes de programación de entorno servidor

---

## ► Aplicaciones CGI y derivados:

- Una de las formas más simples de generar páginas dinámicas es delegar la creación de las mismas a un programa externo, que reciba ciertos parámetros de entrada y devuelva como resultado el contenido que debe visualizar el cliente
- La funcionalidad básica de un servidor web es proveer de contenido estático a un cliente que ha realizado una petición a través de un navegador.
- Puesto que el programa externo no depende del código a generar, el lenguaje elegido puede ser cualquiera (como C o C++)
- El estándar CGI (Common Gateway Interface), especificado en la RFC3875, define este comportamiento.
- Desventaja: escaso rendimiento a la hora de responder a múltiples peticiones CGI simultáneamente. Alternativas:
  - Único proceso que atiende múltiples peticiones. FastCGI.
  - Servidor independiente capaz de ejecutar programas java (Servlets, JavaBeans, ...)
  - EJB / JavaBeans.

# Lenguajes de programación de entorno servidor

---

- ▶ **Aplicaciones híbridas de código repartido:**
  - ▶ Tecnología intermedia alternativa a los lenguajes de scripting (interpretados) y a las aplicaciones CGI y derivadas.
  - ▶ Solución más representativa: plataforma Microsoft .Net Framework a través de ASP.Net.
    - ▶ Los formularios web están contenidos en archivos con una extensión ASPX que son los que el cliente solicita a través de una URL al servidor.
    - ▶ Estos ficheros ASPX contienen código HTML o estático y también etiquetas propias de la plataforma .net.
    - ▶ Estas etiquetas definen Controles Web que se procesan del lado del servidor y Controles de Usuario donde los desarrolladores colocan todo el código estático y dinámico requerido por la página web.

# Integración Servidores Web

---

- ▶ Para ello el cliente debe realizar una petición formal al servidor:
  - ▶ Direcciones mediante URL (Localizador Uniforme de Recursos) que implica una dirección IP
  - ▶ Referencia a un cierto protocolo: HTTP, HTTPS, FTP, ...
  - ▶ Posible ruta hacia el objeto que queremos rescatar.
  - ▶ Posible también especificar un puerto para la comunicación con el servidor.
- ▶ Métodos de comunicación cliente/servidor:
  - ▶ GET. Solicita al servidor que le devuelva la información identificada en la propia URL.
  - ▶ POST. Se usa habitualmente para enviar información al servidor. Normalmente al enviar contenido en un formulario.

# Herramientas de Programación

---

- ▶ El proceso de desarrollo no tiene por qué realizarse en el mismo equipo en el que finalmente se despliegue y ejecute la aplicación web que se está desarrollando.
- ▶ Instrumentos involucrados en el desarrollo Web:
  - ▶ Navegadores.
  - ▶ Editores de documentos.
  - ▶ Entornos de programación.
  - ▶ Herramientas de tratamiento de imágenes.
  - ▶ Herramientas para la creación y administración de bases de datos.
- ▶ Entornos de programación: programas, aplicaciones o simples utilidades destinadas a la programación web.

# Herramientas de Programación

---

## ► Entornos de Programación:

### ► Marcadores de texto:

- Nos ayudan a identificar mejor la sintaxis del lenguaje, cambiando de color las etiquetas, realizando tabulaciones en el texto, etc.
- Arachnophilia, Notepad++, UltraEdit, Context, Sublimetext, Atom etc.

### ► Herramientas genéricas:

- Ofrecen capacidades tales como la sugerencia de estructuras o funciones predeterminadas o la posibilidad de validar la corrección del código escrito.
- Microsoft FrontPage, Eclipse, Dreamweaver, etc.

### ► Herramientas específicas:

- Herramientas específicas para una tecnología concreta, que exigen que el desarrollador tenga instalada una plataforma de desarrollo concreta.
- Microsoft Visual Studio, NetBeans IDE, etc.