

## TEMA 1.- SELECCIÓN DE ARQUITECTURAS Y HERRAMIENTAS DE PROGRAMACIÓN.

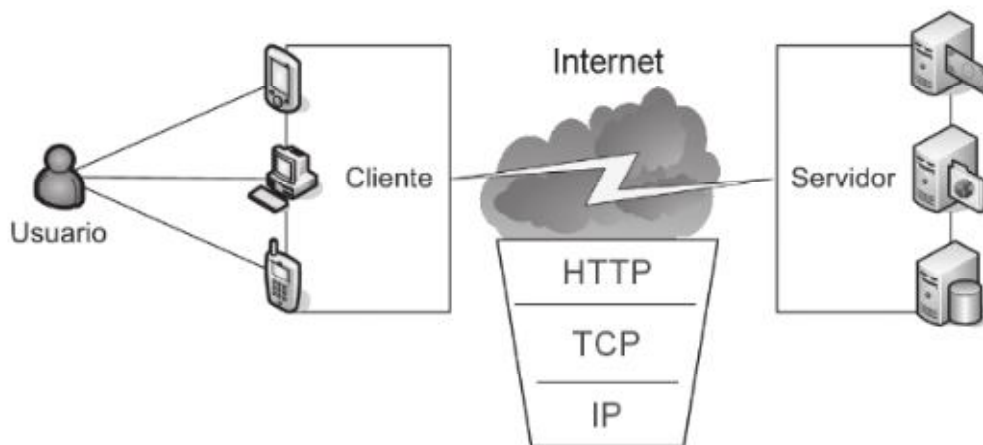
### Objetivos

- Diferenciar los modelos de ejecución de código en entorno cliente/servidor.
- Conocer los mecanismos de ejecución de código en función de la tecnología web.
- Reconocer las ventajas de la generación dinámica de páginas web.
- Identificar los principales lenguajes de programación del entorno del servidor.
- Conocer las distintas herramientas de programación web para lenguajes de servidor.

### Contenidos

#### 1.1.- Modelo de programación en entornos Cliente/Servidor.

El desarrollo en entornos web la arquitectura tecnológica más habitual es la basada en el modelo cliente/servidor, en el que el cliente solicita información mientras que el componente software que responde a esa solicitud se conoce como servidor.



Las funcionalidades en los entornos cliente/servidor se suelen agrupar en capas:

- **Capa de Presentación:** es la capa que ve el usuario. Se centra en el formateo de la información enviada por el servidor y la captura de acciones realizadas por el usuario.
- **Capa de negocio:** es la capa que realiza la funcionalidad de la aplicación.
- **Capa de persistencia o de datos:** es la capa donde residen los datos y la encargada del acceso a los mismos.

#### 1.2.- Generación dinámica de páginas web.

Las aplicaciones web en función de la capacidad de comunicación con el usuario se clasifican en:

- **Aplicaciones Web Estáticas.** Son las páginas en las que la interacción del usuario no genera ningún tipo de acción. Normalmente el lenguaje utilizado es HTML.
- **Aplicaciones Web Dinámicas.** Son aquellas en las que la interacción del usuario provoca algún cambio en la visualización. Normalmente los lenguajes utilizados son HTML, CSS y Javascript.

- **Aplicaciones Web Interactivas.** La interacción hace que se genere un diálogo entre el cliente y el servidor. Normalmente los lenguajes utilizados en el lado del cliente son Activex, Applets de Java y Ajax, mientras que en el lado del servidor son PHP, JSP, ASP, CGI, Servlets.

### 1.3.- Lenguajes de Programación en Entorno Servidor.

En los lenguajes de programación de entorno servidor el código es interpretado o precompilado por un software en el servidor.

Existen diferentes alternativas a la hora de ejecutar código en el servidor:

- Intercalando el código en el HTML de la página web que será interpretada por el servidor y enviada al cliente. Lenguajes como PHP y ASP.
- Enlaces a componentes ejecutables. Lenguajes como CGI y JSP.
- Estrategias híbridas basadas en las llamadas técnicas de respaldo. Lenguajes como ASP.Net

### Lenguajes de Scripting.

Es el lenguaje que se intercala en el código HTML de la aplicación web. La idea es crear una plantilla HTML en la que se inserta el código del lenguaje que es ejecutado por el servidor determinando la parte dinámica de la página web.

Existen diversos lenguajes de scripting:

- **PHP (Hypertext Processor):** Es el más extendido, es gratuito, abierto, portable y ejecutable en diferentes plataformas. Se utiliza en servidores Apache.
- **ASP (Active Server Pages):** Tecnología propietaria de Microsoft y código cerrado. Se utiliza en servidores IIS (Internet Information Server) de Microsoft.
- **Perl.** Fue uno de los primeros lenguajes utilizados para crear aplicaciones CGI's. Es de código abierto.
- **Python.** Es gratuito, multiplataforma y orientado a objetos.
- **JSP (Java Server Pages).** El módulo encargado de ejecutar la página, cuando llega al código Java, lo transforma en un servlet y lo ejecuta. Mejora rendimiento.

### Aplicaciones CGI y derivados.

Los CGI's son programas externos que reciben ciertos parámetros de entrada y devuelven como resultado el contenido que debe visualizar el cliente.

Los CGI's ralentizan el servidor y se intentó mejorar con la tecnología FastCGI.

### Aplicaciones híbridas de código repartido.

ASP.Net de Microsoft es una tecnología que puede ser escrita en cualquier lenguaje del entorno .Net Framework (VB.Net, C#, JScript.Net). Las páginas en ASP.Net son conocidas como Web Forms.

#### 1.4.- Integración con los servidores web.

Existen diferentes métodos para intercambiar información entre el cliente y el servidor:

- **Método GET:** La solicitud de información viene identificada en la propia URL.
- **Método POST:** En la solicitud de información los datos enviados no son visibles por el usuario.

#### 1.5.- Herramientas de Programación.

Clasificación en función de sus capacidades:

- **Navegadores.** Permiten la visualización de las páginas web y la interacción con el usuario.
- **Editores de Documentos.** Editores de texto que permiten escribir código sin ningún tipo de ayuda.
- **Entornos de Programación.** Entornos integrados que permiten editar, compilar y ejecutar los lenguajes usados en el desarrollo de la página web.
- **Herramientas de tratamiento de imágenes.** Permiten adecuar el contenidos gráfico que se utilizará en la web.
- **Herramientas para creación y administración de bases de datos.** Permiten cargar y mantener los datos almacenados en el servidor.

#### Marcadores de Texto.

Nos ayudan a identificar la sintaxis del lenguaje cambiando el color de las etiquetas, cerrando etiquetas abiertas, indicando la sintaxis, tabulando el código.

Por ejemplo: Notepad++, Sublime Text, Brackets,...

#### Herramientas Genéricas.

Tienen funciones más avanzadas que los marcadores de texto como la validación de código, sugerencias de funciones y estructuras, ...

Por ejemplo: Eclipse, Frontpage, Dreamweaver,...

#### Herramientas Específicas.

Herramientas específicas para el desarrollo de aplicaciones de un determinado lenguaje.

Por ejemplo: Microsoft Visual Studio, Netbeans IDE,...

**Práctica 1.** Realizar un resumen de los servidores Apache e IIS con sus principales características, ventajas e inconvenientes. (2 páginas aprox)