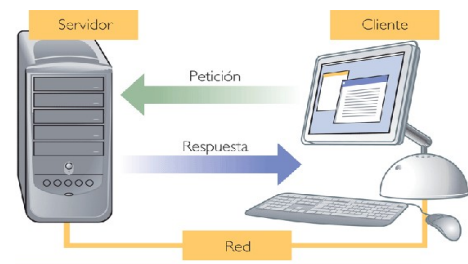


1. Arquitecturas de programación de aplicaciones Web.

1. Modelos de programación Cliente-Servidor.

Las aplicaciones Web se basan en el protocolo HTTP y sus derivados – como HTTPS -, de tal forma que los usuarios acceden a través de un cliente Web – normalmente un navegador – y se conectan al servidor que contiene la aplicación, estableciendo una conversación que consiste en peticiones y respuestas. El protocolo HTTP está basado a su vez en el modelo Cliente-Servidor, de tal forma que el servidor de aplicaciones Web permanecerá a la espera de peticiones por parte de los clientes, y responderá a éstas peticiones enviando el contenido generado – ya sea en formato HTML, XML, etc -. En la terminología técnica, se dice que el cliente es un consumidor de servicios mientras que el servidor es un proveedor de servicios.



En los entornos Cliente-Servidor se distinguen las siguientes funcionalidades o capas:

- Capa de presentación: es la que interactúa directamente con el usuario, es decir, el interfaz de usuario.
- Capa de negocio: es la que controla la lógica de la aplicación, es decir, implementa las funcionalidades de la aplicación.
- Capa de persistencia o de datos: es la que contiene la información de la aplicación y le permite acceder a ellos.

Los modelos de programación Cliente-Servidor se pueden clasificar siguiendo distintos criterios:

- Según el tamaño de sus componentes:
 - Fat Server (Thin Client): El mayor peso de la aplicación recae en el servidor, reservando para el cliente sólo la presentación.
 - Fat Client (Thin Server): El cliente es el que soporta la mayor carga de la aplicación.
- Según el servicio ofrecido: pueden ser servidores web, servidores de ficheros, servidores de correo, servidores de Bases de Datos, etc.
- Según el reparto de funciones entre el cliente y el servidor:
 - Modelo en dos capas: el cliente soporta la capa de presentación y el servidor la de negocio. En este caso, la capa de persistencia no se considera porque o bien no se implementa o va incluida en la de negocio.
 - Modelo en tres capas: En este caso la capa de presentación se implementa en

el cliente, y las de negocio y persistencia se implementan en el servidor. Puede darse el caso de que ambas se encuentren físicamente en la misma máquina o que estén en distintos servidores, interconectados. Además, lo más común es que la capa de persistencia se implemente usando un Sistema Gestor de Bases de Datos (SGBD).

2. Generación dinámica de páginas Web.

Para empezar, es importante distinguir tres tipos de páginas Web:

- Páginas Web estáticas: son las páginas más clásicas, que contienen información estática y consisten en ficheros HTML almacenadas en un servidor Web. La única interacción que permiten al usuario es la navegación a través de hiperenlaces.
- Páginas Web dinámicas: son páginas Web que permiten cierta interacción al usuario, normalmente enfocada a la presentación. Consisten, como en el caso anterior, en ficheros HTML, que en éste caso además contienen scripts que se ejecutan en el navegador - javascript, vbscript, flash – y que proporcionan la interacción. A ésta combinación se le suele llamar DHTML – de Dynamic HTML -.
- Aplicaciones Web: en éste caso se trata de una aplicación que se ejecuta en el servidor y que genera dinámicamente las páginas Web en respuesta a las peticiones del cliente. El contenido de las páginas generadas dependerá de la petición y del estado de la aplicación. En general, las acciones del cliente provocarán cambios en el estado de la aplicación, estableciendo una verdadera interacción. Éstas serán nuestro objeto de estudio.

3. Tecnologías de programación en entorno servidor.

Existen distintas tecnologías de programación y ejecución de aplicaciones Web, y distintos lenguajes asociados. Éstas tecnologías tienen en común que todas se ejecutan en un entorno de servidor Web, de tal forma que la aplicación no tiene que implementar los detalles de la conexión, sino que reciben las peticiones, previamente procesadas por el servidor, y generan la respuesta – normalmente en formato HTML - , que será devuelta al servidor, que a su vez, se la enviará al cliente.

Podemos distinguir tres tipos de tecnologías de programación de servidores:

- CGI (Common Gateway Interface): El servidor delega la creación de páginas dinámicas a una aplicación externa un - programa ejecutable - , pasándole a su entrada estándar los parámetros de la petición, y tomando, de su salida estándar, la página generada. Es el modelo más antiguo de programación Web, y su principal desventaja es el escaso rendimiento a la hora de atender múltiples peticiones simultáneas – ya que crea un proceso nuevo por cada petición -.

- Tecnologías basadas en lenguajes de scripting: integran fragmentos de programa – scripts – dentro del código HTML de las páginas, usando etiquetas específicas. Normalmente se usan lenguajes interpretados. Cuando el servidor recibe una petición, carga la página y la procesa, pasando directamente a la respuesta las líneas de código HTML estático y ejecutando el código incluido en los scripts para generar el contenido dinámico. Algunas de las tecnologías de scripting más usadas son:
 - JSP (Java Server Pages). Usa unos servidores especializados, llamados contenedores JSP, que procesan cada página la primera vez que es requerida, generando clases Java – conocidas como servlets – que serán las encargadas de generar el contenido dinámicamente. Estos servlets se compilan y serán ejecutados para cada petición, en un ambiente de Máquina Virtual Java (JVM) integrado en el servidor. Ésta tecnología mejora el rendimiento para peticiones simultáneas. El lenguaje usado en los scripts es Java, junto con algunas extensiones. Esto le confiere una gran capacidad de integración con otros sistemas y le proporciona una gran cantidad de bibliotecas de programación.
 - PHP (PHP Hypertext Preprocessor). Se basa en un lenguaje interpretado, relativamente sencillo y potente. Se ejecuta sobre un servidor Web al que se le debe incluir una extensión específica para PHP. Tiene una gran aceptación en el mercado.
 - ASP.Net. Está basada en la plataforma .Net de Microsoft y usa los lenguajes propios de ésta. Se trata de una solución híbrida – semi-interpretada -. Su principal inconveniente es que sólo se integra con las tecnologías de Microsoft – servidores Internet Information Server -.

4. Herramientas de programación.

Debido a la naturaleza de las aplicaciones Web, en las que intervienen diversas tecnologías, es normal que se usen también distintos tipos de herramientas. Entre ellas podemos encontrar:

- Herramientas de diseño gráfico.
- Herramientas de diseño Web: BlueFish, DreamWeaver ...
- Editores de texto: Notepad++, gedit ...
- Entornos de Desarrollo – IDE -: NetBeans, Eclipse, Microsoft Visual Studio ...