

WEB INTERACTIVA

La web hipertexto no permite más interacción del usuario que la de seguir hiperenlaces para obtener contenido estático. La web interactiva permite que este contenido no sea estático.

Como primer ejemplo tenemos los **formularios**, el funcionamiento básico de estos es el siguiente:

1. El cliente solicita una página al servidor web
2. El servidor web envía una página que contiene un formulario
3. El cliente envía el formulario al servidor con los datos “suministrados” por el usuario.
4. El servidor recibe y procesa la petición, identifica los recursos del formulario, extrae los parámetros e invoca la funcionalidad correspondiente
5. El programa genera una página con los resultados de su ejecución y la devuelve al cliente a través del servidor web

Los formularios pueden ser GET y POST (entre otros), las diferencias entre estos son que si enviamos un formulario mediante un GET, la información será visible, mientras que si la enviamos mediante un POST la información no será visible.

SESIONES Y COOKIES

Una **sesión** es la relación que se establece entre un cliente y un servidor durante un tiempo finito.

HTTP es un protocolo sin memoria por lo tanto se usan las **cookies**, estos son pequeños fragmentos de información que el servidor almacena en el cliente (con su permiso).

CGI

En el caso de que un servidor quiera obtener información de otro programa, este puede usar un CGI (Common Gateway Interface). El CGI recibe la información mediante variables de entorno, líneas de comandos ...

Los CGI son muy fáciles de programar y pueden usar cualquier lenguaje de programación, pero son lentos y el programa termina con cada llamada, por lo tanto, no es posible el mantenimiento automático de un estado.

WEB API: WEB APPLICATION PROGRAMMING INTERFACES

Las **Web API** surgen con el fin de sustituir a las CGI, debido a su bajo rendimiento. Las Web API se enlazan junto al servidor en una librería dinámica, el servidor llama a las funciones de librería como tareas dentro del propio proceso servidor, este proceso servidor no finaliza ya que se mantienen ficheros abiertos, conexiones a bases de datos ...

Los inconvenientes de las Web API son que, si hay un fallo en una rutina, el servidor puede caer, los lenguajes están limitados, y es difícil de programar.

INTERFACES HÍBRIDAS (WEB API y CGI)

Estas interfaces tratan de conseguir las ventajas de ambas implementaciones evitando sus inconvenientes.

Estas interfaces se desarrollan de modo independiente al servidor web. Durante la inicialización del servidor se pueden arrancar programas en procesos independientes, estos programas se quedan a la espera de recibir peticiones, y tras atender una petición, el programa se queda a la espera de una nueva petición.

PÁGINAS DINÁMICAS

Existe otra alternativa a las Web API y los CGI, estas son las **páginas dinámicas**. Son extensiones del lenguaje HTML que permiten una mayor capacidad de procesamiento.

En el cliente hay código en el documento el cual interpretará para variar dinámicamente la presentación de la página.

En el servidor está el código funcional del fichero HTML el cual será interpretado por el servidor para generar dinámicamente la página antes de su envío al cliente.