

Unidad 2

Desarrollo de aplicaciones Web.

R. A. 2.1

Selecciona la plataforma de desarrollo para aplicaciones web, de acuerdo con la capacidad y funcionalidad diseñada para el sitio Web.

Cómo elegir una plataforma de desarrollo para crear una página web

A. Determinación del software de desarrollo.

El HTML puede considerarse el lenguaje nativo de Internet, pero no es el único que podemos encontrar hoy en día. Cada vez existe un mayor número de lenguajes de programación, tanto para web como para móviles, que nos permiten ampliar las posibilidades de desarrollo de una manera más sencilla.

Los diferentes lenguajes utilizados son Python, JavaScript, PHP y Ruby.

A la hora de seleccionar con qué lenguaje trabajar, es importante conocer las diferencias y singularidades de cada uno de ellos, sus ventajas e inconvenientes, en función de la plataforma para la que estemos desarrollando nuestro proyecto. Además, también conviene familiarizarse con el concepto framework, cada vez más extendido, y que hace referencia al conjunto de buenas prácticas, módulos y librerías de cada lenguaje para abordar proyectos y necesidades concretas.



Programa que realiza algunas tareas en beneficio de otras aplicaciones llamadas clientes

Un servidor de aplicaciones es una máquina remota, en este caso alojada en el SCI, y sobre la que al conectarme puedo ejecutar aplicaciones sin necesidad de tenerlas instaladas en mi propio equipo.

Es útil cuando los recursos de mi ordenador no sean lo suficientemente potentes como para poder instalarla de forma local o cuando tengo un sistema operativo distinto e incompatible con el software que necesito.

Requerimientos.

Análisis: especifica que es lo que la aplicación debe hacer.

Diseño: Establece como alcanzar el objetivo.

El análisis de requerimientos comienza conociendo el problema o la idea principal, tarea que define el alcance y la naturaleza de la idea o necesidad a cubrir. Seguido de la indagación por medio de entrevistas y, finalmente, la documentación de todo lo recopilado que será el entregable de este trabajo. El objetivo de realizar la indagación de los requerimientos es proporcionar a todas las partes un entendimiento escrito del problema. El cual se logra por medio de varios productos del trabajo: casos de uso, diagramas de flujo y de características, modelos de requerimientos o especificaciones, entre otros.

B. Selección del servidor de aplicaciones

Los servidores de aplicaciones actúan como intermediarios entre las peticiones de los usuarios de una página web y los servidores de bases de datos y de transacciones, situándose en la segunda capa de una arquitectura de red trinivel. De esta forma se consigue aumentar las funcionalidades ofrecidas en un sitio de Internet sin necesidad de actualizar los programas cliente utilizados por los usuarios.

Para elegir el servidor de aplicaciones a utilizar se deben tener en cuenta múltiples factores, que podemos clasificar en las siguientes categorías:

- Seguridad
- Escalabilidad y fiabilidad
- Soporte de las transacciones
- Presentación de páginas web

Servidores de Base de Datos. Son los que se utilizan para la ejecución de gestores de BD y que varios usuarios hagan operaciones sobre ellas al mismo tiempo, situado en un lugar diferente; se puede acceder a las BD por terminales o equipos con un programa llamado cliente que permita el acceso a ellas.

El servidor MySQL fue desarrollado originalmente para manejar grandes bases de datos mucho más rápido que las soluciones existentes y ha estado siendo usado exitosamente en ambientes de producción sumamente exigentes por varios años. Aunque se encuentra en desarrollo constante, el servidor MySQL ofrece hoy un conjunto rico y útil de funciones. Su conectividad, velocidad, y seguridad hacen de MySQL un servidor bastante apropiado para acceder a bases de datos en Internet.

Servidor local “Virtual host”

Es un servidor que reside en una red local al equipo de referencia.

El servidor web local puede estar instalado en cualquiera de los equipos que forman parte de una red local.

Se instala por medio de un programa para permitirnos probar y navegar la página web que vayamos a crear desde nuestro equipo.



HTTP:

Es el protocolo que permite que los navegadores y servidores web se comuniquen adecuadamente. Este es utilizado por navegadores web para solicitar archivos HTML de parte de los servidores remotos. Así, los usuarios podrán interactuar con dichos archivos mediante la visualización de las páginas web que cuentan con imágenes, música, vídeos, texto, etc.

El protocolo HTTP tiene como base a TCP, el cual implementa un modelo de comunicación cliente-servidor.

El protocolo HTTP tiene como base a TCP, el cual implementa un modelo de comunicación cliente-servidor. Existen tres tipos de mensajes que HTTP utiliza:

- HTTP GET
- HTTP POST
- HTTP HEAD

No debemos olvidar el protocolo HTTPS, el cual nos proporciona seguridad punto a punto (entre el cliente y el servidor web). El protocolo HTTPS utiliza el protocolo TLS (Transport Layer Security) que también utiliza TCP por encima.

HTTP vs HTTPS



Desarrolla aplicaciones web, mediante el uso de componentes de la plataforma de desarrollo

A. Uso del lenguaje de programación para el desarrollo web



COOKIES: ¿QUÉ SON?

Son pequeños archivos que se instalan en el navegador de los visitantes de una página, con o sin su consentimiento. Sirven para enviar información de ida y vuelta entre el navegador del ordenador y el sitio web visitado.



NO AFECTADAS

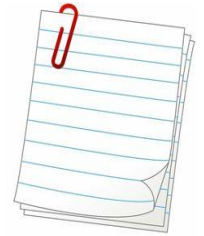
Necesarias para la correcta navegación del usuario en el sitio web, pues permite que la página reconozca el idioma del usuario y el tamaño de la letra, entre otras preferencias. Además, impiden los ataques maliciosos.



AFECTADAS

Permiten medir la actividad del usuario y elaborar estadísticas de su navegación. Asimismo, gestionan la frecuencia y el contenido de los anuncios pero, sobretodo, son importantes porque guardan toda esta información para poder enviarle posteriormente a su salida del sitio, publicidad de nuestro sitio para que lo recuerden, vuelvan a él y realicen una conversión deseada.

Una cookie es un fragmento de información que un navegador web almacena en el disco duro del visitante a una página web. La información se almacena a petición del servidor web, ya sea directamente desde la propia página web con JavaScript o desde el servidor web mediante las cabeceras HTTP, que pueden ser generadas desde un lenguaje de web scripting como PHP.



La información almacenada en una cookie puede ser recuperada por el servidor web en posteriores visitas a la misma página web

Creando cookies

Al recibir una solicitud HTTP, un servidor puede enviar un encabezado Set-Cookie con la respuesta. La cookie generalmente es almacenada por el navegador, y luego la cookie se envía con solicitudes hechas al mismo servidor dentro de un encabezado HTTP Cookie. Se puede especificar una fecha de vencimiento o duración, después de lo cual ya no se envía la cookie. Además, se pueden establecer restricciones a un dominio y ruta específicos, lo que limita el lugar donde se envía la cookie.

El encabezado de respuesta HTTP Set-Cookie envía las cookies del servidor al agente de usuario. Una cookie simple se establece así:

Set-Cookie: <nombre-cookie>=<valor-cookie>



ANTES DE UTILIZARLAS, PREGÚNTATE: ¿ME SON NECESARIAS?

En caso de utilizarlas, es recomendable informar a tus visitantes que vas a instalarlas en su equipo, para qué las utilizarás y brindarles la opción de rechazo.

EXPLICACIÓN DE LA ESTRUCTURA CON HTML 5

HTML 5

CSS : "Los estilos"

- 1** `<div id="contenedor">`
Estilos aplicados a la etiqueta: `<div>`, llamada **contenedor**
Este `<div>` es una especie de *división* que nos permite insertar elementos dentro de ellos, de esta manera dividimos en secciones o "apartados" la página web. En el ejemplo, dentro de este `<div>` estará contenida TODA la página. Observemos que el `<div>` se cierra, hasta el final antes de `</body>`. Es por eso que todo el contenido de nuestra página estará **dentro** de este importante `<div>`, podríamos decirte *el principal*. Los `<div>` también sirven para colocar otros elementos como: imágenes, tablas, listas, formularios, etc, en secciones diferentes.
`#contenedor{`
`width:90%;`
`margin:auto;`
`background-color:#7D7D7D;`
`}`
- 2** `<header id="encabezado">`
El `<header>` es una etiqueta de HTML5 que permite incrustar algunos elementos tales como encabezados, títulos o logos. En el ejemplo, este encabezado, dice "Mi experiencia en la cuarentena", observa como, con los estilos se le colocó color de fondo amarillo.
`#encabezado{`
`text-align:center;`
`background-color:#FFD80E;`
`align-self:center;`
`}`
- 3** `<nav id="menu">`
La etiqueta `<nav>` es una etiqueta de HTML5 que te permite colocar dentro de ella, los **hipertextos** en la parte superior de tu página, para crear la navegación del sitio web. Siempre que coloquemos esta etiqueta, inmediatamente después dentro de ella debes escribir los enlaces, con la etiqueta `<a href>`
`#menu{`
`background-color:#646464;`
`text-align:center;`
`}`
- 4** `<section id="derecha">`
La etiqueta `<section>`, es una etiqueta semántica de HTML5 que permite, incrustar en ella diversos elementos o contenidos de una página web. En este ejemplo dentro de `<section>` se colocaron: imágenes, párrafos y líneas que permiten tener contenido **para leer** por parte del usuario. Observa como dentro de `<section>` ya se puede hacer uso de `` para imágenes, de `<p>`, para párrafos, de `<hr>` para insertar una línea. Las etiquetas `<figure>` funcionan en este ejemplo, para poder centrar en tu página web imágenes que sean colocadas dentro de `<section>`. La etiqueta `<section>` tiene, en este ejemplo, un "id", como identificador llamado "derecha", el cual dentro de la hoja de estilo CSS, observa como se coloca el estilo: `float:right`, con este estilo aplicado, inmediatamente `section` se ubica en la parte DERECHA.
`#derecha{`
`float:right;`
`background-color:#84B3A8;`
`width:80%;`
`}`
- 5** `<aside id="izquierda">`
La etiqueta `<aside>`, es una especie de contenedor, parecido a `<section>`, que permite incrustar elementos como enlaces, párrafos, publicidad, que no son estrictamente "necesarios" colocarlos, es decir una página html 5 puede no tener el elemento `<aside>`. En este caso se ha colocado para que comprendas un poco más su uso y practicar "la maquetación", la cual te permite "diseñar", con diversas secciones dentro de tu página web, (de este modo lo conoces y practicas). En algunas páginas web, el `<aside>` lo utilizan, comúnmente, para hacer anotaciones personales o pensamientos del autor de la web. En este ejercicio se te pide nos pliques o hables sobre la experiencia personal de la cuarentena. Observa, como en `<aside>`, en el ejemplo, hay hipertextos que te permiten ir otras páginas web. Dentro de CSS, se coloca: `float:left`, para ubicar este elemento a la IZQUIERDA.
`#izquierda{`
`background-color:#646464;`
`float:left;`
`width:20%;`
`text-align:center;`
`}`
- 6** `<footer id="pie pagina">`
La etiqueta `<footer>` es una etiqueta de HTML5 que te permite colocar un pie de página al final de tu documento web. "Foot" en español significa pie-, es comúnmente usado para poner iconos de redes sociales o información de contacto de la empresa o del diseñador que ha construido la página web.
`#piepagina{`
`clear:both;`
`background-color:#8F0884;`
`}`

B. Verificación del funcionamiento y funcionalidad de la aplicación

La planificación de la verificación debe definir el propósito, los métodos, las frecuencias y las responsabilidades de las actividades de verificación. Los resultados de la planificación deben estar en un formato adecuado para su registro.

Verificación: Es un conjunto de actividades que aseguran que el software implementa correctamente una función específica, esto implica comprobar que el software corresponde con lo especificado.

La validación de datos garantiza la corrección de datos y precisión de todos los valores de datos de la aplicación

El objetivo de estos pasos es asegurar que el sistema satisface las expectativas y procura demostrar que el software hace lo que se espera.

Maquetación

En esta etapa se organiza todos los elementos de la futura página web, se estructura de manera gráfica los contenidos escritos, imágenes, video, títulos, enlaces, menús, botones, formularios, y todo aquel elemento que haya sido considerado en la lista de requerimientos del proyecto.

Estrictamente el proceso de maquetar, sólo se refiere al acto de estructurar la página web, distribuir y organizar los elementos y los espacios de la página. Sin embargo, este proceso se mejora cuando la maquetación se convierte en una pre-visualización del sitio web, es decir, además de estructurar y organizar los elementos, también se elige una paleta de colores, y se diseña ya sea en papel o en algún editor gráfico la visión final del sitio web.

