APLICACIONES WEB: CLIENTE-SERVIDOR

Tarea 5 - DOO

Las Aplicaciones web son todas aquellas herramientas que se ejecutan al acceder a un servidor web mediante un navegador.

Es decir que los datos o los archivos en los que trabajas son procesados y almacenados dentro de la web.

Alguna de las ventajas de las aplicaciones web es la facilidad de actualizarlas sin necesidad de instalar software en los clientes.

En esta tarea en importante el primero comprender como funcionan las aplicaciones web porque de aquí se basa la comunicación cliente-servidor. Cuando hablamos de aplicaciones web no nos referimos a una aplicación necesariamente instalada en nuestra computadora si no que se encuentra en un servidor muy lejos de nuestro equipo y este funciona a través de la internet por lo que muchos clientes se conectaran a este servidor para utilizar las funciones de la aplicación.

Gracias al lenguaje de programación Java y otros más es posible que muchos clientes trabajen con esta aplicación, sin que exista el riesgo de que llegue a saturarse dicho uso, ya que el sistema de programación de estos sistemas trabaja de manera paralela para con todo aquel cliente que ingresa hasta dicho servidor.

¿Y ahora como hacemos que el cliente se sienta seguro y con confianza en la página web?

Haciendo uso del método *HTTPS*, (Protocolo de transferencia de hipertexto) es un protocolo de comunicación de Internet que protege la integridad y la confidencialidad de los datos de los usuarios entre sus ordenadores y el sitio web. Puesto que el usuario espera que su experiencia online sea segura y privada.

El envío de datos mediante el protocolo *HTTPS* está protegido mediante el protocolo de seguridad de la capa de transporte (TLS), que proporciona las tres capas clave de seguridad siguientes:

Cifrado: se cifran los datos intercambiados para mantenerlos a salvo de miradas indiscretas.

Integridad de los datos: los datos no pueden modificarse ni dañarse durante las transferencias.

Autenticación: garantiza que tus usuarios se comuniquen con el sitio web previsto. Proporciona protección frente a los ataques "man-in-the-middle" y contribuye a la confianza de los usuarios, lo que se traduce en otros beneficios empresariales.

Para ayudar al servidor a diferenciar un cliente de otro, cada cliente debe identificarse a sí mismo con el servidor. El rastreo de clientes individuales, conocido como rastreo de sesiones, puede lograrse de varias formas.

Una técnica popular utiliza cookies; otra utiliza el objeto HttpSession. Otras técnicas adicionales de rastreo de sesiones incluyen el uso de elementos input form de tipo "hidden" y la reescritura de URLs. Con los elementos "hidden", un formulario Web puede escribir los datos de rastreo de sesión en un componente form en la página Web que devuelve al cliente, en respuesta a una petición previa.

Sesiones

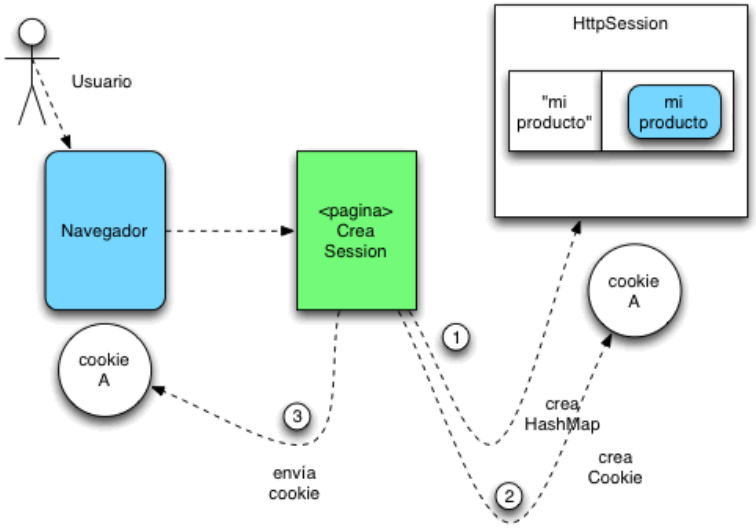
Una sesión es una serie de comunicaciones entre un cliente y un servidor en la que se realiza un intercambio de información. Por medio de una sesión se puede hacer un seguimiento de un usuario a través de la aplicación.

Para poder hacer uso de las sesiones hay que poner el atributo session de la directiva page a true de la siguiente manera:

<%@page session=’true’%>

Las sesiones que se crean se llevan a cabo mediante la interface HttpSession que que sirve para almacenar información entre diferentes peticiones HTTP ya que crea nuevas conexiones con el servidor web cuando el cliente produce una peticion.

Cada vez que un usuario crea una session accediendo a una página se crea un objeto a nivel de Servidor con un HashMap(una coleccion de objetos) vacío que nos permite almacenar la información que necesitamos relativa a este usuario. Realizado este primer paso se envía al navegador del usuario una Cookie que sirve para identificarle y asociarle el HashMap que se acaba de construir para que pueda almacenar información en él.



COOKIE

Una cookie es una pequeña información enviada por un sitio web y almacenada en el navegador del usuario, de manera que el sitio web puede consultar la actividad previa del usuario. La computadora recibe una cookie la primera vez que entra a un sitio web y se reactiva cada vez que el usuario vuelve a visitar dicha página para personalizar futuras visitas del usuario al sitio web.

Una cookie no debe contener información directa del usuario ni recolectar información del equipo del usuario.

¿Como funciona?

Una cookie consiste en una cadena de texto (string) con varios pares key=value cada uno separado por **;**.

```
<nombre>=<valor>; expires=<fecha>; max-age=<segundos>; path=<ruta>; domain=<dominio>;
```

Desde el servidor, las cookies son creadas mediante la cabecera de respuesta HTTP Set-Cookie.

```
Set-Cookie: NOMBRE=VALOR; domain=NOMBRE_DOMINIO; expires=FECHA DE VENCIMIENTO DE LA COOKIE
```

Pares de valores que pueden ser usados en una cookie:

Atributo	Valor	Sintaxis	Descripción
NOMBRE _DE_COOKIE	VALOR	El nombre y el valor no pueden contener los siguientes caracteres: punto y coma (;), coma (,) o espacio ( ). Dichos valores sólo se pueden agregar utilizando una codificación URL<a/>.	Este es el único atributo **obligatorio**.
expires	FECHA	Día, DD-MM-AAAA HH:MM:SS GMT	El atributo de *expires* se usa para definir la fecha en la que la cookie ya no debe almacenarse en el disco rígido ni ser admitida por el servidor.
domain	nombre_de_dominio	xxx.xxx.xxx	El nombre del dominio generalmente se deja vacío ya que el nombre del servidor se asigna por defecto (generalmente es lo que se quiere aquí). Donde se indique, el nombre del dominio debe contener al menos dos puntos (es decir: www.commentcamarche.net). Un equipo proveniente de un dominio específico sólo puede especificar un nombre de sub-dominio o su propio nombre de dominio

Controles Hidden Field

El control HiddenField constituye un medio para almacenar información en la página sin mostrarla.

Un ejemplo seria el elemento input de tipo "hidden" donde permite incluir datos que no pueden ser vistos o modificados por el usuario cuando se envia un formulario.

La información incluida en un control HiddenField no se muestra cuando el explorador representa la página, aunque los usuarios pueden ver el contenido del control si consultan el código fuente de la página. Por tanto, no debe guardar

información confidencial en ningún control HiddenField, como los Id. de usuario, las contraseñas o información sobre la tarjeta de crédito.

En conclusión, tenemos que las sesiones son muy importantes en el mundo el comercial del internet, por ejemplo, en las tiendas virtuales online pues se mejora la experiencia del usuario ya que permite vincular la información del visitante a lo largo de sus diversos a la aplicación web.

Y esto se logra a través de cookies, algunas ventajas de utilizar cookies es que nosotros le damos la duración que solicitemos pueden ser días, meses o años. Aparte de los posibles problemas de privacidad que puede ocasionar el uso de cookies, también hay que tener en cuenta que la manipulación de las cookies creadas por nuestra aplicación web se puede convertir en un arma de ataque contra nuestra propia aplicación, por lo que debemos ser extremadamente cuidadosos a la hora de decidir qué datos almacenarán las cookies.

Bibliography

Deitel, P. &. (2008). *Como programar en Java.* Mexico: Pearson Educación de México, S.A. de C.V. .

Microsoft. (n.d.). Retrieved from https://msdn.microsoft.com/es-es/library/ms227988(v=vs.100).aspx

Mozilla. (n.d.). Retrieved from https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input/hidden

Tomcat. (n.d.). Retrieved from https://tomcat.apache.org/tomcat-5.5-doc/servletapi/javax/servlet/http/HttpSession.html