UT5 - DESARROLLO DE APLICACIONES WEB UTILIZANDO CÓDIGO EMBEBIDO

Desarrollo Web en Entorno Servidor

2° DAW

Autenticación y control de acceso

- Es importante verificar la identidad de los dos extremos de una comunicación.
- Se debería utilizar el protocolo HTTPS.

- En la mayoría de las aplicaciones web existe un mecanismo de control de acceso que obligue al usuario a identificarse.

Mecanismos de autentificación

- El protocolo HTTP ofrece un método sencillo para autentificar a los usuarios:
  - El servidor web debe proveer algún método para **definir los usuarios** que se utilizarán y cómo se pueden autentificar
  - Cuando un **usuario no autentificado intenta acceder** a un recurso restringido, el **servidor web responde con un error** de "Acceso no autorizado" (código 401).
  - El **navegador** recibe el error y **abre una ventana** para solicitar al usuario que se autentifique mediante su **nombre y contraseña**
  - La información de autentificación del usuario **se envía al servidor**, que **la verifica** y decide si permite o no el acceso al recurso solicitado. Esta **información se mantiene en el navegador** para utilizarse en posteriores peticiones a ese servidor

Mecanismos de autentificación

- En Apache existe la utilidad **htpasswd**
  - Fichero con usuarios y contraseñas
  - Crearlo en un lugar no accesible por los usuarios del servidor web
- Pero ¿cómo le indicamos a Apache qué recursos tienen acceso restringido?
  - Fichero .htaccess
- Además tendrás que asegurarte de que en la configuración de Apache se utiliza la directiva **AllowOverride** para que se aplique correctamente la configuración que figura en los ficheros .htaccess.

Mecanismos de autentificación

- Desde PHP puedes acceder a la información de autentificación HTTP que ha introducido el usuario utilizando el array superglobal **$_SERVER**
  - **$_SERVER['PHP_AUTH_USER']** Nombre de usuario que se ha introducido.
  - **$_SERVER['PHP_AUTH_PW']** Contraseña introducida.
  - **$_SERVER['AUTH_TYPE']** Método HTTP usado para autentificar. Puede ser Basic o Digest

Mecanismos de autentificación

- En PHP puedes usar la función **header** para forzar a que el servidor envíe un error de "Acceso no autorizado" (código 401). De esta forma no es necesario utilizar ficheros **.htaccess** para indicarle a Apache qué recursos están restringidos
- En su lugar, puedes añadir las siguientes líneas en tus páginas PHP:

```
if (!isset($_SERVER['PHP_AUTH_USER'])) {
    header('WWW-Authenticate: Basic Realm="Contenido restringido"');
    header('HTTP/1.0 401 Unauthorized');
    echo "Usuario no reconocido!";
    exit();
}
```

Mecanismos de autentificación

- Te habrás dado cuenta en el ejercicio anterior que ahora se solicitan credenciales HTTP, pero el servidor no verifica la información. Deberemos ser nosotros los que lo comprobemos.

```
if ($_SERVER['PHP_AUTH_USER']!='usuario' ||
     $_SERVER['PHP_AUTH_PW']!='contraseña') {
     header('WWW-Authenticate: Basic Realm="Contenido
restringido"');
     header('HTTP/1.0 401 Unauthorized');
     echo "Usuario no reconocido!";
     exit();
}
```

- **Pregunta:**
  - ¿Creéis que esto es correcto? ¿Veis algún inconveniente? ¿Se os ocurre alguna alternativa?

Mecanismos de autentificación

- Una solución mejor es utilizar un almacenamiento externo para los nombres de usuario y sus contraseñas. Para esto podrías emplear un fichero de texto, o mejor aún, una base de datos. La información de autentificación podrá estar aislada en su propia **base de datos**, o compartir espacio de almacenamiento con los datos que utilice tu aplicación web.

Hoja05_Sesiones_01 (Ej 1)

- **Pregunta**
  - ¿Cómo almacenaríais la contraseña?

Mecanismos de autentificación

- **MD5** es un método para generar un resumen de un texto o un documento, de tal forma que a partir del resumen obtenido no es posible recuperar el texto original, ni hallar otro texto a partir del cual se obtenga el mismo resumen. Se llama hash al resumen obtenido al aplicar una función hash. Una de las funciones hash más extendidas es MD5, que genera 128 bits como resumen (normalmente se representa mediante una cadena de texto de 28 caracteres o mediante 32 dígitos hexadecimales)
- En PHP puedes usar la función **md5** para calcular el hash MD5 de una cadena de texto

```
$str = 'contraseña';
if (md5($str) ==
'4c882dcb24bcb1bc225391a602feca7c') {
    echo "Contraseña correcta";
}
```

Hoja05_Sesiones_01 (Ej 2, 3 y 4)

Ejercicio de entrega (I)

- Antes de comenzar a ver el manejo de sesiones realizaremos un ejercicio que debéis entregar.
- El trabajo se realizará en grupo.
- La idea es construir una aplicación web que sea una tienda. Cada grupo elegirá la temática de su tienda.
- Diseñaréis la base de datos en la que al menos tiene que haber una tabla de usuarios y otra de productos
  - Debéis crear una página **registro.php** para que se registren usuarios (ya hecha)
  - En otra debéis mostrar un listado de los productos (**productos.php**) y para cada uno de ellos debe existir un botón "Añadir a la cesta"

    Lo más sencillo es que creéis un formulario por cada producto

    Más adelante completaremos esta página

Ejercicio de entrega (I)

Listado de productos

Cesta de la compra

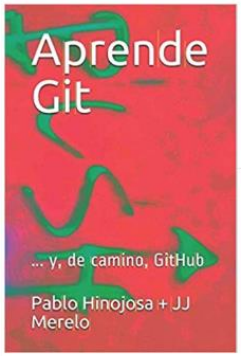
Aprende SQL en un fin de semana: El curso definitivo para crear y consultar bases de datos

Bases de datos

4.99 €

Descripción del producto Reseña del editor ¡Oferta de lanzamiento de la edición impresa! Sólo 4,99 € por tiempo limitado. El curso de SQL definitiv...

Añadir a la cesta



Aprende Git: ... y, de camino, GitHub

Programación y desarrollo de software

4.90 €

git es un sistema de control de versiones distribuido, que dicho así suena geek y aburrido, pero que en la práctica es una forma de trabajar en equipo...

Añadir a la cesta

Pentesting con Kali: Aprende a dominar la herramienta Kali de pentesting, hacking y auditorías activas de seguridad

Seguridad informática

29.93 €

Aprende la profesión de pentester, y a dedicarte al hacking ético. Kali es una distribución de Linux que contiene centenares de herramientas para h...

Añadir a la cesta

Java para novatos

Programación y desarrollo de software

15.26 €

Todo lo que necesitas saber para empezar a programar en Java aplicando el paradigma de orientación a objetos desde el primer momento. ¿Te han dicho...

Añadir a la cesta

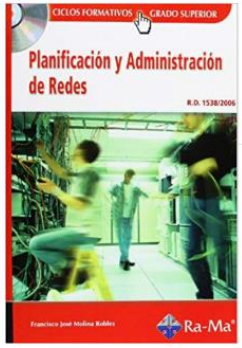
No me hagas pensar

Medios digitales y diseño gráfico

18.95 €

Cientos de miles de diseñadores y desarrolladores web se han basado en la guía del gurú de usabilidad Steve Krug para ayudarles a entender los princip...

Añadir a la cesta



Planificación y Administración de Redes

Redes y administración de sistemas

37.90 €

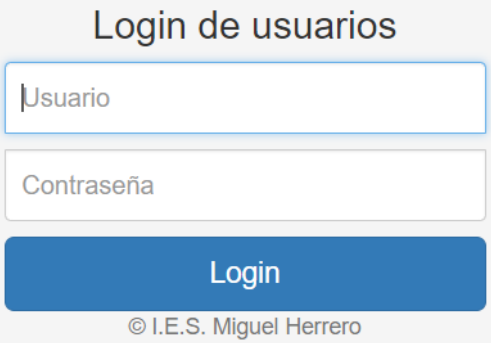
La presente obra está dirigida a los estudiantes del Ciclo Formativo de Grado Superior de Administración de Sistemas Informáticos en Red, en concreto ...

Añadir a la cesta

Ejercicio de entrega (II)

- **Login.php**
  - Vas a crear una página **login.php** con un formulario con dos campos, uno de tipo text para el usuario, y otro de tipo password para la contraseña.
  - Al pulsar el botón Enviar, el formulario se enviará a esta misma página, donde se compararán las credenciales proporcionadas por el usuario con las almacenadas en la base de datos.
  - Cuando un usuario proporciona unas credenciales de inicio de sesión correctas, se le redirige de forma automática a la página del listado de productos (**productos.php**)

Ejercicio de entrega (II)



Cookies

- Pero, ¿qué es una **cookie**?
- Una cookie es un fichero de texto que un sitio web guarda en el entorno del usuario del navegador.
  - Su uso más típico es el almacenamiento de las preferencias del usuario (por ejemplo, el idioma en que se deben mostrar las páginas), para que no tenga que volver a indicarlas la próxima vez que visite el sitio

Hoja05_Sesiones_02 (Ej 1)

Cookies

- En PHP, para almacenar una cookie en el navegador del usuario, podemos utilizar la función **setcookie**
- **Pregunta**
  - ¿Qué hará la siguiente instrucción?
    - setcookie("nombre_usuario", $_SERVER['PHP_AUTH_USER'], time()+3600);
  - ¿Os parece que es aconsejable guardar esta información en cookies?

- El proceso de recuperación de la información que almacena una cookie es muy simple. Cuando accedes a un sitio web, el navegador le envía de forma automática todo el contenido de las cookies que almacene relativas a ese sitio en concreto. Desde PHP puedes acceder a esta información por medio del array **$_COOKIE**

Hoja05_Sesiones_02 (Ej 2)

Gestión de sesiones

- El término sesión hace referencia al conjunto de información relativa a un usuario concreto.
  - Ejemplos:
    - Nombre del usuario
    - Artículos de la lista de la compra de una tienda online
- Cada usuario distinto de un sitio web tiene su propia información de sesión.
- Para distinguir una sesión de otra se usan los identificadores de sesión (**SID**).
- Un **SID** es un atributo que se asigna a cada uno de los visitantes de un sitio web y lo identifica

Gestión de sesiones

- **Pregunta**
  - ¿Dónde se almacena ese SID?
- Existen dos maneras de mantener el SID entre las páginas de un sitio web que visita el usuario:
  - Utilizando cookies, tal y como vimos
  - Propagando el SID en un parámetro de la URL. El SID se añade como una parte más de la URL, de la forma:
    - *http://www.misitioweb.com/tienda/listado.php&PHPSESSID=3453 4fg4ffg34ty*

Gestión de sesiones

- En PHP el manejo de sesiones está automatizado en gran medida.
  - Cuando un usuario visita un sitio web, no es necesario programar un procedimiento para ver si existe un SID previo y cargar los datos asociados con el mismo. Tampoco tienes que utilizar la función **setcookie** si quieres almacenar los SID en cookies, o ir pasando el SID entre las páginas web de tu sitio si te decides por propagarlo. Todo esto PHP lo hace automáticamente.
- Por defecto, PHP incluye soporte de sesiones incorporado.

- Sin embargo, antes de utilizar sesiones en tu sitio web, debes configurar correctamente PHP utilizando las siguientes directivas en el fichero *php.ini* según corresponda

Gestión de sesiones

Directiva	Significado
session.use_cookies	Indica si se deben usar cookies (1) o propagación en la URL (0) para almacenar el SID.
session.use_only_cookies	Se debe activar (1) cuando utilizas cookies para almacenar los SID, y además no quieres que se reconozcan los SID que se puedan pasar como parte de la URL (este método se puede usar para usurpar el identificador de otro usuario)
session.save_handler	Se utiliza para indicar a PHP cómo debe almacenar los datos de la sesión del usuario. Existen cuatro opciones: en ficheros (files), en memoria (mm), en una base de datos SQLite (sqlite) o utilizando para ello funciones que debe definir el programador (user). El valor por defecto (files) funcionará sin problemas en la mayoría de los casos.
session.name	Determina el nombre de la cookie que se utilizará para guardar el SID. Su valor por defecto es PHPSESSID.

Gestión de sesiones

Directiva	Significado
session.auto_start	Su valor por defecto es 0, y en este caso deberás usar la función **session_start** para gestionar el inicio de las sesiones. Si usas sesiones en el sitio web, puede ser buena idea cambiar su valor a 1 para que PHP active de forma automática el manejo de sesiones.
session.cookie_lifetime	Si utilizas la URL para propagar el SID, éste se perderá cuando cierres tu navegador. Sin embargo, si utilizas cookies, el SID se mantendrá mientras no se destruya la cookie. En su valor por defecto (0), las cookies se destruyen cuando se cierra el navegador. Si quieres que se mantenga el SID durante más tiempo, debes indicar en esta directiva ese tiempo en segundos.
session.gc_maxlifetime	Indica el tiempo en segundos que se debe mantener activa la sesión, aunque no haya ninguna actividad por parte del usuario. Su valor por defecto es 1440. Es decir, pasados 24 minutos desde la última actividad por parte del usuario, se cierra su sesión automáticamente

Gestión de sesiones

- El inicio de una sesión puede tener lugar de dos formas:
  - Si has activado la directiva **session.auto_start** en la configuración de PHP, la sesión comenzará automáticamente en cuanto un usuario se conecte a tu sitio web
  - Si no se utiliza el inicio automático de sesiones, habrá que ejecutar la función **session_start** para indicar a PHP que inicie una nueva sesión o reanude la anterior. Está función devuelve **false** en caso de no poder iniciar o restaurar la sesión
- Mientras la sesión permanece abierta, puedes utilizar la variable superglobal **$_SESSION** para añadir información a la sesión del usuario, o para acceder a la información almacenada en la sesión.

Hoja05_Sesiones_03 (Ej 1 y 2)

Ejercicio de entrega (III)

- **Login.php**
  - Modifica la página de login hecha en sesiones anteriores para utilizar sesiones.
  - Si los datos son correctos, se iniciará una nueva sesión y se almacenará en ella el nombre del usuario que se acaba de conectar.
- **Productos.php**
  - Tanto en esta página como en todas las demás, es necesario comprobar la variable de sesión $_SESSION['usuario'] para verificar que el usuario se ha autentificado correctamente. Si el usuario no se ha autentificado, se muestra un mensaje de error junto con un enlace a la página **login.php**
  - También debemos mostrar el **nombre de usuario** en la cabecera de la página. Este dato lo sacaremos de la sesión.
  - El botón "Añadir" envía a esta misma página los datos código, nombre y precio del producto.
  - Cuando se abre la página, se comprueba si se ha enviado este formulario, y si fuera así se añade un elemento al array asociativo **$_SESSION['cesta']** con los datos del nuevo producto. El array $_SESSION['cesta'] es la variable de sesión en la que guardaremos los datos de todos los productos que va a comprar el usuario.

Gestión de sesiones

- Para eliminar la información almacenada en la sesión:
  - **session_unset**. Elimina las variables almacenadas en la sesión actual, pero no elimina la información de la sesión del dispositivo de almacenamiento usado. Sería similar a hacer $_SESSION = array();
  - **session_destroy**. Elimina completamente la información de la sesión del dispositivo de almacenamiento.

Hoja05_Sesiones_03 (Ej 3)

Ejercicio de entrega (IV)

- **Productos.php**
  - En esta página se mostrará el número de productos añadidos a la cesta y el importe total de ésta.
  - También habrá dos formularios: Uno para vaciar la cesta (botón "Vaciar Cesta"), dirigido a esta misma página, y otro para realizar la compra (botón "Comprar"), que dirige a la página **cesta.php**
  - Contendrá un botón para desconectar al usuario actual. Llama a la página **logoff.php**, que borrará la sesión actual.

Ejercicio de entrega (IV)

Listado de productos

3 productos **(53.78 €)**

Vaciar Cesta | Comprar

Desconectar usuario ivan

Aprende SQL en un fin de semana: El curso definitivo para crear y consultar bases de datos

Bases de datos

4.99 €

Descripción del producto Reseña del editor ¡Oferta de lanzamiento de la edición impresa! Sólo 4,99 € por tiempo limitado. El curso de SQL definitiv...

Añadir a la cesta

Aprende Git: ... y, de camino, GitHub

Programación y desarrollo de software

4.90 €

git es un sistema de control de versiones distribuido, que dicho así suena geek y aburrido, pero que en la práctica es una forma de trabajar en equipo...

Añadir a la cesta

Pentesting con Kali: Aprende a dominar la herramienta Kali de pentesting, hacking y auditorías activas de seguridad

Seguridad informática

29.93 €

Aprende la profesión de pentester, y a dedicarte al hacking ético. Kali es una distribución de Linux que contiene centenares de herramientas para h...

Añadir a la cesta

Java para novatos

Programación y desarrollo de software

15.26 €

Todo lo que necesitas saber para empezar a programar en Java aplicando el paradigma de orientación a objetos desde el primer momento. ¿Te han dicho...

Añadir a la cesta

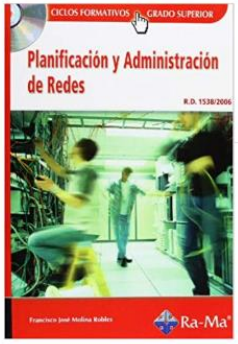
No me hagas pensar

Medios digitales y diseño gráfico

18.95 €

Cientos de miles de diseñadores y desarrolladores web se han basado en la guía del gurú de usabilidad Steve Krug para ayudarles a entender los princip...

Añadir a la cesta



Planificación y Administración de Redes

Redes y administración de sistemas

37.90 €

La presente obra está dirigida a los estudiantes del Ciclo Formativo de Grado Superior de Administración de Sistemas Informáticos en Red, en concreto ...

Añadir a la cesta

Ejercicio de entrega (IV)

- **Cesta.php**
  - Se muestra un resumen de los productos que ha seleccionado junto con el importe de los mismos.
  - Los datos que figuran en la página se obtienen todos de la información almacenada en la sesión del usuario. No es necesario establecer conexiones con la base de datos.
  - En esta página habrá dos formularios que simplemente redirigen a otras páginas: "Pagar", que redirige a la página **pagar.php**, que en nuestro caso lo único que debe hacer es eliminar la cesta del usuario. Y el que contiene el botón de desconexión, que es similar al que figuraba en la página productos.php, y dirige a la página **logoff.php**, que cierra la sesión del usuario

Ejercicio de entrega (IV)

Cesta de la compra

3 productos **(53.78 €)**

Vaciar Cesta | Pagar

Desconectar usuario ivan

Aprende Git: ... y, de camino, GitHub

4.90 €

Pentesting con Kali: Aprende a dominar la herramienta Kali de pentesting, hacking y auditorías activas de seguridad

29.93 €

No me hagas pensar

18.95 €

Seguir comprando | Pagar

Ejercicio de entrega (V)

- **Logoff.php**
  - Tanto desde la cesta como desde la página del listado de productos, se le ofrece al usuario la posibilidad de cerrar la sesión. Para ello se le dirige a la página logoff.php, que no muestra nada en pantalla y su única función es recuperar la sesión, eliminarla y redirigir a **login.php**
- **Pagar.php**
  - Simplemente se recupera la información de la sesión y la elimina.