

Practica 5: Programación Web Dinámica

Sergio Paracuellos, Ricardo Muñoz

Índice

1. Introducción	4
2. Objetivos	4
3. Enunciado	4
4. Especificaciones	4
4.1. Entorno hardware	4
4.2. Entorno software	5
5. Tecnología empleada para la generación de páginas web dinámicas	5
6. Escenario de web dinámica	5
7. Proceso de instalación y configuración	6
7.1. Instalación de Php	6
7.2. Integración Php y apache	6
7.3. Instalación de MySql	6
7.3.1. my.cnf	6
7.3.2. Acceso a la base de datos	6
7.4. Integración Php y MySql	7
8. Código	7
8.1. Página principal (index.php)	7
8.2. Acceso mediante login (login.php)	8
8.3. Implementación de Sesiones	9
8.4. Test de Sesiones(test.php)	9
8.5. Implementación de logout(logout.php)	9
8.6. Implementación de Cookies	10
8.7. Página de Registro de un Nuevo Usuario (registro.php)	10
8.8. Registro en base de datos (registrar.php)	10
8.9. Página Privada: Zona de descarga de pacticas (practicass.php)	11

Índice de figuras

1.	Página principal	8
2.	Acceso mediante login	9
3.	Implementación de Sesiones	9
4.	Test de sesiones	9
5.	Logout	9
6.	Implementación de cookies	10
7.	Registro de un nuevo usuario	10
8.	Registrar en la base de datos	11
9.	Zona de descarga de prácticas	11

1. Introducción

Antes de comenzar con el comentario del desarrollo de las prácticas queremos introducir en esta sección como hemos planificado las tareas previas para llevar a cabo este proceso.

Hemos montado un servidor de **cv**s para poder realizar la tarea con control de las distintas versiones y poder trabajar cómodamente cada uno en nuestra casa. Hemos modificado el comportamiento de este **cv**s para que cada vez que se produzca un cambio nos lo notifique enviandonos un mail a una lista de correo que hemos montado para este propósito.

También queremos comentar la notación que vamos a seguir de cara a la documentación que nos ocupa.

- **Negrita**: escribiremos en negrita los comandos del sistema, así como el *seteo de variables* y la correspondiente modificación de los archivos del sistema.
- **Cursiva**: escribiremos en cursiva toda función referente a un script, palabras inglesas o similares.

También queremos destacar que aunque las prácticas hayan sido realizadas en un ordenador portatil, hemos instalado en una partición separada una debian estable desde cero y no sobrepasando en ningún caso el tamaño de un disco ZIP. Como editor hemos utilizado **vi**, para no estar en ningún tipo de desventaja respecto a los otros alumnos que realizan las practicas en la sala con un disco ZIP.

Después de estas consideraciones prerealización de las practicas y documentación, pasaremos a analizar cada una de las prácticas propuestas para este bloque.

2. Objetivos

El objetivo de esta práctica es muy claro: creación de un sitio web con contenidos dinámicos. Estos contenidos serán generados mediante la programación en el lado del servidor.

3. Enunciado

Creación de un sitio web con contenidos dinámicos. Estos contenidos serán generados en el servidor basándose en determinada información, como por ejemplo, datos remitidos por el usuario, el navegador utilizado para realizar la petición HTTP...

4. Especificaciones

4.1. Entorno hardware

Las características de la maquina de prueba son:

- Procesador Pentium III 1.1 GHz
- Tarjeta de red
- Conexion Telefónica ADSL
- Memoria RAM 512 MB

4.2. Entorno software

Los componentes software que posee la máquina de pruebas son:

- Sistema Operativo: GNU/Linux
- Distribución Debian en su version estable 3.0 (woody)
- Servidor Web: Apache v1.3
- Lenguaje Php v4
- SGBD MySql v3.23

5. Tecnologia empleada para la generación de páginas web dinámicas

El lenguaje de programación empleado para la creación del sitio web ha sido **php**, un lenguaje de programación interpretado en el lado del servidor. Sus características más importantes es que se trata de un lenguaje independiente de la plataforma, rápido y con librerías de funciones bastante amplias y muy buena documentación, por lo que se aprendizaje es bastante sencillo y rápido.

6. Escenario de web dinámica

En esta práctica se ha creído conveniente crearnos un sitio web relacionado con las prácticas de la asignatura. Se trata de un sistema donde es necesaria la autenticación mediante password de un usuario que quiera acceder a una zona donde descargarse las prácticas realizadas en el curso. Hemos integrado un Sistema Gestor de Bases de Datos (SGBD) para guardar la información necesaria de los usuarios registrados y tener así un control exhaustivo de la aplicación web. Este sitio web utiliza mecanismos de seguimiento del estado del cliente tanto con el empleo de sesiones con su correspondiente implementación en el sistema de Logout para que el cliente cierre su sesión cuando lo desee, como el empleo de cookies (sesiones permanentes) donde reconocerá al cliente que ha visitado la página con anterioridad. Además de todo esto, se ha implementado el envío de correo electrónico cuando un usuario es registrado en la página, dándole la bienvenida e informándole de su actual nick y password.

7. Proceso de instalación y configuración

7.1. Instalación de Php

Para la instalación de php lo hacemos a partir de binarios de la rama estable de debian. El paquete que debemos instalar es *php4*. Simplemente haremos *apt-get install php4*.

7.2. Integración Php y apache

Los binarios proporcionados por el propio apache y php en la rama estable de Debian hoy por hoy incorporan unos perfectos scripts de integración de tal forma que todo lo necesario para que esto funcione correctamente nos lo dan hecho. No obstante podemos mirar el fichero de configuración de apache y darnos cuenta que cargamos el módulo de php con el correspondiente *LoadModule*.

7.3. Instalación de MySql

Para la instalación de nuestro SGBD necesitamos 3 paquetes de software:

- `mysql-client`
- `mysql-server`
- `mysql-common`

7.3.1. `my.cnf`

7.3.2. Acceso a la base de datos

Una vez en este punto, debemos entrar en la base de datos para crear la tabla o tablas necesarias que utilizará nuestra aplicación web. Los pasos a seguir para crear la base de datos y sus tablas son:

- Ejecutar **`mysql -u root -p`**.
- Ahora nos pedirá la contraseña pues metemos la contraseña correspondiente. Se puede modificar (`man mysql`).
- Una vez dentro de mysql debemos crear la base de datos. La llamamos `sanri` así es que ejecutamos: **`create database sanri;`**
- Especificaremos a continuación que se quiere emplear la base de datos `sanri` para crear tablas: **`use sanri;`**
- Crear la tabla 'users' ejecutando el siguiente comando: **`create table users (usuario varchar(20), pass varchar(30), mail varchar(20));`**
- Si todo se ha ejecutado sin problemas ya tenemos lista la base de datos para que sea utilizada por nuestra aplicación.

7.4. Integración Php y MySql

Para la integración de php y mysql debemos instalar otro paquete de software. En este caso se trata de **php4-mysql** cuya versión debería coincidir con la del paquete php4 y así nos evitaríamos conflictos. Lo realizamos de manera análoga a como hemos ido comentando en otras prácticas *apt-get install php4-mysql*. Ya hemos integrado php4 y mysql ya que los scripts del propio paquete que Debian proporciona nos solucionan cualquier problema que nos haya podido ocurrir.

8. Código

8.1. Página principal (index.php)

```

<?
session_start()
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">
<html>
<head>
</head>
<body bgcolor=#AABCDD>
  <h1>PRACTICA 4:</h1>
  <ul>
    <li><a href=primero.html>Primer Enlace: </a>(Accesible desde una maquina
    del laboratorio)</li>
    <li><a href=segundo.html>Segundo Enlace: </a>(Accesible desde la subred
    del laboratorio)</li>
    <li><a href=tercero.html>Tercer Enlace: </a>(Dominio publico excepto una
    maquina del laboratorio)</li>
    <li><a href=cuarto.html>Cuarto Enlace: </a>(Autenticacion)</li>
  </ul>
  <br>
  <h1>PRACTICA 5:</h1>
  <?
    if (isset($_HTTP_COOKIE_VARS["cookiep"])) {
      echo "Hola " . $cookiep. "!!!";
    }
  ?>
  <p>Entrar!</p>
  <table>
  <tr>
    <form method="post" action='login.php'>
    <td>Nick:</td>
    <td><input type="text" name="nick" size="20"><br></td>
  </tr>
  <tr>
    <td>Password:</td>
    <td><input type="password" name="password" size="20"><br></td>
  </tr>
  </table>
  <input type="radio" name="tipo" value="persistente" checked> Cookies
  Persistentes<br>
  <input type="radio" name="tipo" value="session"> Cookies Session<br><br>

```

8.2 Acceso mediante login (login.php)

```
<input type="submit" value="login" name="validar" size="8">
</form>
<?
    if((session_is_registered("cookie"))) {
        print("<form method='post' action='logout.php'>");
        print("<input type='submit' value='logout' name='validar' size='8'>");
        print("</form>");
    }
?>
```

50

```
<p> Si todavia no eres usuario registrado, <a
href=registro.php>registrate</a></p>
<p><a href=test.php>Test session</a></p>
</body>
</html>
```

Figura 1: Página principal

8.2. Acceso mediante login (login.php)

```
<?
if(($nick=="") || ($password=="")) {
    print("<p style='text-align:center'>Debes rellenar todos los
campos.<br><br>");
    print("<a href='index.php'>Volver al menu principal</a></p>");
} else {
    $db=mysql_connect("localhost","root","");
    mysql_select_db("sanri", $db);
    $result=mysql_query("SELECT pass FROM users WHERE usuario='$nick'", $db);
    if($row=mysql_fetch_array($result)) {
        $password2=md5($password);
        if($row["pass"]==$password) {
            if($tipo=="session") {
                session_start();
                $cookie=$nick;
                session_register("cookie");
                echo "COOKIE SESSION";
            } else {
                setcookie("cookie", $nick, time()+7200);
                echo "COOKIE PERSISTENTE";
            }
            include("practicas.php");
        } else {
            print("<p style='text-align:center'>Password Incorrecto.<br><br>");
            print("<a href='index.php'>Volver al menu principal</a></p>");
        }
        mysql_free_result($result);
    } else {
        sleep(2);
        print("<p style='text-align:center'>El nick no existe<br><br>");
    }
}
?>
```

Figura 2: Acceso mediante login

8.3. Implementación de Sesiones

Podemos observar el código de *login.php* en el cual veremos lo siguiente:

```
<?
if($tipo=="session") {
    session_start();
    $cookie=$nick;
    session_register("cookie");
    echo "COOKIE SESSION";
}
?>
```

Figura 3: Implementación de Sesiones

8.4. Test de Sesiones(test.php)

```
<?
    session_start();
    if(session_is_registered("cookie")) {
        echo "La cookie esta puesta y su valor es: ".$cookie;
    } else {
        echo "No esta la cookie";
    }
?>
```

Figura 4: Test de sesiones

8.5. Implementación de logout(logout.php)

```
<?
    session_start();
    session_destroy();
?>
<p style='text-align:center'>La sesion ha sido destruida.<p>
<p style='text-align:center'><a href="index.php">Volver</a></p>
```

Figura 5: Logout

8.6. Implementación de Cookies

Dentro de *login.php* podemos ver:

```
<?
setcookie("cookiep", $nick, time()+7200);
echo "COOKIE PERSISTENTE";
?>
```

Figura 6: Implementación de cookies

8.7. Página de Registro de un Nuevo Usuario (registro.php)

```
<!DOCTYPE html public "-//W3C//DTD HTML 4.0 //EN">
<html>
<head><title> REGISTRO </title></head>
<body>
<p> Rellena este formulario para registrarte:</p>
<table>
  <tr>
    <td><form method="post" action="registrar.php">
      <td>Nick: </td>
      <td><input type="text" name="usuario"><br></td>
    </tr>
    <tr>
      <td>Password:</td>
      <td><input type="password" name="pass"><br></td>
    </tr>
    <tr>
      <td>Correo electronico</td>
      <td><input type="email" name="mail"><br></td>
    </tr>
  </table>
  <p><input type="submit" value="Registrar" name="validar">
  <input type="reset" value="Borrar">
</form>
</body>
</html>
```

Figura 7: Registro de un nuevo usuario

8.8. Registro en base de datos (registrar.php)

```
<body>
<?
  if(($usuario=="") || ($pass=="") || ($mail=="")) {
    print("<p style='text-align:center'>Debe rellenar todos los campos.
    <br><br>");
    print("<a href='registrar.php'>Volver</a></p>");
  }
```

8.9 Página Privada: Zona de descarga de pacticas (practicass.php)

```
    } else {
        $db=mysql_connect("localhost", "root", "");
        mysql_select_db("sanri", $db);
        $result=mysql_query("SELECT * FROM users WHERE usuario='$usuario'", $db); 10
        if($row=mysql_fetch_array($result)) {
            print("<p style='text-align:center'> El nick ya esta en uso, seleccione
            otro<br><br>");
            print("<a href='registrar.php'>Volver</a></p>");
        } else {
            $password2=md5($pass);
            $sentencia="INSERT INTO users (usuario, pass, mail) VALUES ('$usuario',
            '$pass', '$mail')";
            mysql_query($sentencia, $db);
            mail($mail, "registro", "Ha sido registrado"); 20
            print("<p style='text-align:center'>Ha sido registrado<br><br>");
            print("<a href='index.php'> Volver </a></p>");
        }
    }
?>
</body>
```

Figura 8: Registrar en la base de datos

8.9. Página Privada: Zona de descarga de pacticas (practicass.php)

```
<body bgcolor=#AADDDB3>
<h1>PRACTICAS SANRI</h1>
<ul>
    <li>Practica 1: Configuracion FTP<br>
    <p> Como configurar tu servidor ftp <a href="descargas/ftp.pdf">PDF</a></p>
    </li>
    <li>Practica 2: Configuracion de Servidor de Correo<br>
    <p> Como configurar tu servidor de correo <a href="descargas/correo.pdf">PDF</a></p>
    </li>
    <li>Practica 3: Configuracion Squid<br> 10
    <p> Como configurar tu servidor Proxy-cache <a href="descargas/squid.pdf">PDF</a></p>
    </li>
    <li>Practica 4: Configuracion Apache<br>
    <p> Como configurar tu servidor web Apache <a href="descargas/apache.pdf">PDF</a></p>
    </li>
    <li>Practica 5: Configuracion Programacion web dinamica<br>
    <p> Como crear una web dinamica con PHP y Mysql<a href="descargas/php.pdf">PDF</a></p>
    </li>
</ul>
</body>
```

20

Figura 9: Zona de descarga de prácticas