



Universidad de Guayaquil
Facultad de Ciencias Matemáticas y Físicas
Carrera de Software
Ciclo I (Online)

Asignatura: Desarrollo de Aplicaciones Web.

Paralelo: SOF-S-MA-6-1.

Fecha: 9 de Agosto del 2021.

Integrantes (Grupo 3):

(Todos los integrantes colaboraron en el trabajo grupal)

- Pincay Chilan Alexis Daniel.
- Tacuri Arizaga Bryan Elian.
- Rodríguez Merchán Darwin Alexander.
- Zambrano Manzano Adriana Brigitte.

Tema: Técnicas de mantenimiento de estado: Cookies y sesiones.

Mientras un usuario visita nuestro sitio o aplicación web él puede visitar varias de sus páginas, las técnicas de mantenimiento de estado nos ayudan a poder mantener información que el usuario genera mientras navega y que podremos acceder en cualquiera de esas páginas. Los portales web reconocen las acciones que el usuario ha realizado anteriormente: si se ha registrado, qué elementos ha visitado, las compras añadidas a la cesta, etc. Y según está información modifican su contenido o funcionalidad. Para conseguir esto, es necesario almacenar esta información temporalmente utilizando ya sean cookies o sesiones. Para poder revisar estos temas se propone el desarrollo de los siguientes enunciados:

Parte 1: Cookies

1. ¿Qué son y dónde se almacenan las cookies?

Una cookie se usa a menudo para identificar a un usuario. Una cookie es un pequeño archivo que el servidor incrusta en la computadora del usuario. Cada vez que la misma computadora solicita una página con un navegador, también enviará la cookie. Con PHP, puede crear y recuperar valores de cookies.

Una cookie es un archivo creado por un sitio de Internet para almacenar información en el equipo, como por ejemplo, tus preferencias cuando visitas ese sitio. Cuando visites un sitio que usa cookies, el sitio puede pedirle a Firefox que guarde una o más cookies en el disco duro. En algunos navegadores, cada cookie es un pequeño archivo, pero en Firefox todas las cookies se guardan en un solo archivo ubicado en la carpeta perfil.

- Indique cuáles y para qué sirven los parámetros que puede recibir el método `setcookie(...)`

Se crea una cookie con la `setcookie()` función.

`setcookie (name, value, expire, path, domain, secure, httponly);`

Solo se requiere el parámetro de nombre. Todos los demás parámetros son opcionales. Con los parámetros 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.

- Escribe la línea de código php para crear una cookie que dure 12 horas y que sea válida para cualquier directorio.

```
<?php
// (43200 es igual a 12 horas) Expira en 12 horas
$cookie_name = "user";
$cookie_value = "Adriana Zambrano";
setcookie ($cookie_name, $cookie_value, time( ) + (43200), "/");
?>
```

- Escribe la línea de código php para crear una cookie que dure 3 días y que sea válida solo para el directorio `/datos/generales`

```
<?php
// (86400 * 3 es igual a 3 días) Expira en 3 días
$cookie_name = "user";
$cookie_value = "Adriana Zambrano";
setcookie ($cookie_name, $cookie_value, time( ) + (86400 * 3), "/datos/generales");
?>
```

- Escriba una aplicación que conste de tres páginas.
 - En la primera página hay un formulario para elegir una película de un combo (coloque al menos 3 títulos de películas) y hay un botón para enviar los datos a la segunda página.
 - En la segunda página se lee el parámetro enviado, se muestra algún texto de sinopsis de la película elegida en la primera página y su imagen, se crea una cookie (método `setcookie`) con el título de la película seleccionada. Adicionalmente hay un enlace a la tercera página.
 - En la tercera página lea la cookie creada (utilizar superGlobal `$_COOKIE`) y muestre su valor.

(Nombre del archivo: ejercicio1)

Parte 2: Sesiones

1. Indique para qué sirven cada uno de los siguientes métodos

a. `session_start()`:

Este método permite crear una sesión o reanudar la que se está usando actualmente en un tipo de identificador de sesión pasado por medio de una petición GET o POST o también mediante una cookie. Cuando este método en mención es llamado o invocado se autoinicia una sesión, al realizar esta acción PHP llamará a los gestores de almacenamiento de sesiones `open` y `read`. La llamada de retorno `read` recuperará cualquier información de sesión existente, se almacenará en un formato serializado especial; y será deserializada y usada para rellenar automáticamente la variable superGlobal `$_SESSION` cuando la llamada de retorno `read` devuelva la información de sesión guardada a la gestión de sesiones de PHP.

b. `session_unset()`:

Esta función es de tipo `void` la cual libera todas las variables de sesión actualmente registradas.

Se hace una advertencia al usar esta función porque al usarla deshabilitará el registro de variables de sesión a través del array superGlobal `$_SESSION`.

c. `session_destroy()`:

Esta función de tipo booleana destruye la información asociada con la sesión actual. No destruye ninguna de las variables globales asociadas con la sesión, ni destruye la cookie de sesión. Para volver a utilizar las variables de sesión se debe llamar a `session_start()`. Si queremos destruir la sesión de forma completa, como desconectar al usuario, el id de la sesión también debe ser destruido. Si utilizamos una cookie para propagar el id de la sesión entonces, la cookie de sesión se debe eliminar. Lo más idóneo sería utilizar `setcookie()`.

2. Realice una aplicación web que le permita al usuario seleccionar uno de entre varios artículos (elijá usted el tipo de artículos). Cada vez que el usuario selecciona un nuevo artículo se almacena en sesión dos variables (utilizar `$_SESSION`): una con el nombre o código del artículo seleccionado y otra con el tiempo de expiración del artículo. En otra página de su aplicación lea las variables de sesión y muestre su información (utilizar `$_SESSION`). A diferencia de las cookies, en las sesiones podemos almacenar no solo cadenas de caracteres sino todo tipo de datos. En el ejercicio a continuación se solicita almacenar en sesión un arreglo de objetos.

(Nombre del archivo: ejercicio2)

3. Crear una aplicación web para la carga de miembros un directorio empresarial. Para lograrlo se crearán 4 archivos PHP, form.php (Formulario), registro.php (Agrega el miembro), mostrar.php (Mostrar datos de sesión) y cerrar.php (elimina la sesión). En todas las páginas se debe iniciar sesión. [session_start()]

a. form.php: De los miembros del directorio se desea registrar su email, nombre, género (combo), no olvidar el botón de envió. El método del formulario será POST y será enviado a la página «registro.php».

b. La página registro.php recibe los datos del formulario, se verifica la llegada y el contenido de los elementos recibidos (utilizar métodos isset y empty). En caso de que los datos llegaron correctamente, se almacenara en un objeto de la clase Miembro (debe tener los atributos email, nombre, genero), y este objeto se agrega a un array de nombre lista. Almacene este arreglo en la sesión (\$_SESSION). Enlace de Ejemplo arreglos de objetos: https://www.youtube.com/watch?v=k7b9Oq5ZO_I

NOTA: Considerar que antes de almacenar un objeto al array lista, se debe verificar si la lista ya está almacenada en la sesión. Si ya existe se debe leer el array lista de la sesión y a éste agregarle el nuevo objeto. Posteriormente volver a guardar el array lista en la sesión.

c. En la página mostrar.php, se imprimirá la lista de miembros de la lista de correo, en una tabla con una columna de numeración, el nombre del miembro, su dirección de email y su género. Adicionalmente existirá un enlace para cerrar sesión (echo "Cerrar sesión"). En cerrar.php se cerrará la sesión (session_destroy()).

(Nombre del archivo: ejercicio3)