Control semana 6: Sesiones y Control de Usuarios con PHP

Jorge Peñaloza

Programación Avanzada I

Instituto IACC

19 de julio de 2021

**Desarrollo**

**Respuestas del problema**

***Contexto***

1. De acuerdo con los conocimientos previos respecto al uso de sesiones en la programación avanzada, describa que características debe tener una sesión.
2. Que tipos de errores podría diferenciar en el uso de sesiones. Fundamente.
3. Cree una tienda virtual para una librería, no más de 5 artículos, combinando el uso de sesiones y carrito de compra.

Se requiere:

* El código debe incluirse en la entrega y además deberán presentarse las capturas de pantallas respectivas. Esto con la finalidad de identificar los posibles errores que se presenten la ejecución de lo solicitado.

***Respuesta a pregunta 1***

Las características que debe poseer una sesión son:

1. Establece un vinculo entre el cliente y el servidor de tal forma que el cliente no necesita estar enviando sus datos cada vez que realiza una petición.
2. Las sesiones se deben iniciar en cada proceso que ejecute una rutina, de tal forma que el servidor tenga acceso de los datos de la sesión creada, este procese se puede ver en *código de programación 1.*
3. Una ves inicializados los datos de una sesión, se creará un identificador de la sesión, ademas los datos de esta sesión pueden utilizar tal como se muestra en *código de programación 2.*

Código de programación 1

*Archivo: index.php*

<?php

session\_start();

$\_SESSION['nombre'] = "Jorge";

$\_SESSION['apellido'] = "Peñaloza";

?>

<a href="sesiones.php">Prueba</a>

Fuente: elaboración propia

Código de programación 2

*Archivo: sesiones.php*

<?php

session\_start();

echo $\_SESSION['nombre'];

echo $\_SESSION['apellido'];

print\_r($SESSION);

?>

Fuente: elaboración propia

***Respuesta a pregunta 2***

Los errores mas comunes que pueden darse en una sesión son lo vistos en la tabla 1, en la cual se indica cual será el comportamiento que existirá en la ejecución del script.

Tabla 1

*Vista de archivos binarios, CSV y de texto en Notepad*

|  |  |  |
| --- | --- | --- |
| Constante | Descripción | Interrupción de Script |
| E\_ERROR | Errores fatales en tiempo de ejecución | SI |
|  |
| E\_WARNING | Advertencias no fatales en tiempos de ejecución | NO |  |
| E\_NOTICE | Avisos en tiempo de ejecución, que indican que el script encontró algo que podría ser un error u ocurrir en curso normal de un script | NO |  |
| E\_STRICT | Sugerencias de cambios al código para ampliar la compatibilidad con versiones posteriores de PHP | NO |  |
| E\_DEPRECATED | Avisos en tiempo de ejecución, sobre funciones obsoletas | NO |  |
| E\_ALL | Todos los anteriores (excepto E\_STRIC, que es recién incluido en E\_ALL, desde la versión 5.4 de PHP) | SI |  |

Fuente: Obtenido de (IACC, 2020)

***Respuesta a pregunta 3***

Para la realización de este ejercicio se realizó lo siguiente

1. Se utilizarán las siguientes tecnologías para el proyecto.
   1. PHP
   2. MySQL
   3. HTML
   4. Bootstrap
2. Los datos de conexión a la base de datos son los siguientes.
   1. Host: localhost
   2. Nombre de usuario: root
   3. Contraseña: (en blanco)
   4. Base de datos: carta
3. Los datos de la tabla *mis\_productos* son:
   1. id: Corresponde a un código interno para cada producto.
   2. name: Corresponde al nombre del título del libro
   3. descripcion: Corresponde al género del libro
   4. price: Corresponde al precio por unidad del libro
   5. created: Corresponde a la fecha de creación del libro en la tabla
   6. modified: corresponde a la ultima fecha en la cual se realizó una modificación en el registro del libro
4. En el archivo *Configuracion.php* de *código de programación 1* se encuentra la conexiona la base de datos.
5. En el archivo *cerrarsesion.php* de *código de programación 2* se encuentra el cierre de la sesión.
6. En el archivo *funciones.php* de *código de programación 3* se encuentran disponibles varias funciones para gestionar las sesiones.
7. En el archivo *validar.php* de *código de programación 4* se encarga de recibir los datos los login y gestionar la creación de la sesión.
8. En el archivo *settings.php* de *código de programación 5* se encarga de definir constantes para ejemplificar el uso de sesiones con un usuario en concreto llamado *jorge* y contraseña *1,* para estos datos se ha creado un hash para el usuario.
9. El archivo *index.php* de *código de programación 6* contiene el login de la pagina web y se puede ver la figura 1.
10. Cuando se ingresa un usuario, esta pagina llama a *validar.php* para crear la sesión en caso de ser un usuario valido, y luego de esto llamara a *panel.php* cuyo script es *código de programación 7.*
11. Cuando el sistema rechaza el usuario, la página mostrada en la figura 2 mostrara *Bienvenido Error de usuario* como indicador de usuario incorrecto.
12. Cuando el sistema acepta al usuario, la página mostrada en la figura 3 mostrara los libros del carro de compras.
13. En *panel.php* exisaten 2 opciones *Inicio* y *Ver carro de compras,* en el ,cual Inicio mantiene la página *panel.php* y *Ver carro de compras* dirige a la página *VerCarta.php* cuyo script es *código de programación 8*  y contiene los elemento agregados de carrito de compras*.*
14. En la figura 4 se puede apreciar la vista de *VerCarta.php* y además se ve que contiene las opciones del punto 10.
15. En el archivo *La-carta.php* de *código de programación 9* se encuentran una clase llama *Cart* que permite con sus métodos ir totalizando, contando y en resumen gestionando los datos del carro de compras se mostraran en *verCarta.php*.
16. En el archivo *funciones.js* de *código de programación 10* se puede apreciar el script utilizado para gestionar *verCarta.php* mediante el script *AcciónCarta.php* de *código de programación 11*
17. El archivo *estilos.css* de *código de programación 12* contiene los estilos de la pagina agregados a los que utiliza Bootstrap.
18. El archivo *mis\_productos.sql* de *código de programación 13* contiene la consulta SQL para crear la tabla necesaria para el proyecto.
19. Este proyecto está basado en los documentos de (IACC, 2020) para las sesiones y en un ejemplo de (Tapia, 2018) para la gestión del carro de compras, todo ayudo a dar la funcionalidad requerida con sus respectivas modificaciones.

Código de programación 1.

*Archivo: Configuracion.php*

<?php

$dbHost = 'localhost';

$dbUsername = 'root';

$dbPassword = '';

$dbName = 'carta';

$db = new mysqli($dbHost, $dbUsername, $dbPassword, $dbName);

if ($db->connect\_error)

{

    die("No hay Conexion con la base de datos: " . $db->connect\_error);

}

?>

Fuente: elaboración propia

Código de programación 2

*Archivo: cerrarsesion.php*

<?php

include "settings.php";

require\_once("funciones.php");

session\_start();

logout();

?>

Fuente: elaboración propia

Código de programación 3

*Archivo: funciones.php*

<?php

/\*Traer los datos recibidos por HTTP POST y retorno el HASH

MD5 de ambos\*/

function get\_post\_data()

{

    $hash = "";

    if(isset($\_POST['usuario']) &&isset($\_POST['psw']))

    {

        $hash = md5($\_POST['usuario'].$\_POST['psw']);

        return $hash;

    }

}

/\*Comparar ambos hashes. Si son idénticos, retorno Verdadero\*/

function validar\_user\_y\_pass()

{

    $user\_hash = get\_post\_data();

    $system\_hash = HASH\_ACCESO;

    if($user\_hash == $system\_hash)

    {

        return True;

    }

    return false;

}

# Destruir sesión

function logout()

{

    unset($\_SESSION);

    $datos\_cookie = session\_get\_cookie\_params();

    setcookie( session\_name(), NULL, time()-999999,

        $datos\_cookie["path"],

        $datos\_cookie["domain"],

        $datos\_cookie["secure"],

        $datos\_cookie["httponly"]);

    header("Location:".PAGINA\_LOGIN);

}

/\*Primero verificar que la variable de sesión login\_date,

e xiste. De ser así, obtener su valor y lo retornarlo. Si no

e xiste, retornará el entero 0\*/

function obtener\_ultimo\_acceso()

{

    $ultimo\_acceso = 0;

    if(isset($\_SESSION['login\_date']))

    {

        $ultimo\_acceso = $\_SESSION['login\_date'];

    }

    return $ultimo\_acceso;

}

/\*

Esta función, retornará el estado de la sesión: sesión

inactiva, retornará False mientras que sesión esta activa,

retornará True. Al mismo tiempo, s e encarga de actualizar la

variable de sesión

login\_date, cuando la sesión se encuentre activa\*/

function sesion\_activa()

{

    $estado\_activo = False;

    $ultimo\_acceso = obtener\_ultimo\_acceso();

    /\*Establecer como límite máximo de inactividad, para mantener

    la sesión activa, m edia hora (o sea, 1800 segundos). De esta

    manera, sumando 1800 segundos a login\_date, se de finiendo

    cual es la marca de tiempo más alta, que puedo per mitir al

    usuario para mantener la sesión activa. \*/

    $limite\_ultimo\_acceso = $ultimo\_acceso + 1800;

    /\*Rea

    lizar la comparación. Si el último acceso del usuario

    hace más de media hora de gracia que se le otorga para

    mantenerle activa la sesión, es mayor a l a marca de hora

    actual, significa entonces que su sesión puede seguir activa.

    Entonces, actualizar la marc a de tiempo, renovándole la

    sesión\*/

    if($limite\_ultimo\_acceso > time())

    {

        $estado\_activo = True;

        # actualizar la marca de tiempo renovando la sesión

        $\_SESSION['login\_date'] = time();

    }

    return $estado\_activo;

}

# Verificar sesión

function validar\_sesion()

{

    if(!sesion\_activa())

{

     logout();

    }

}

?>

Fuente: elaboración propia

Código de programación 4

*Archivo: validar.php*

<?php

include "settings.php";

include "funciones.php";

if(isset($\_POST['usuario']) && isset($\_POST['psw']))

{

    //session\_start();

    if(get\_post\_data())

    {

        $user\_valido = validar\_user\_y\_pass();

        if($user\_valido)

        {

            session\_start();

            $\_SESSION['usuario'] = $\_POST['usuario'];

            $\_SESSION['login\_date'] = time();

            header("Location:".PAGINA\_RESTRINGIDA\_POR\_DEFECTO);

        }

        else

        {

            header("Location:".PAGINA\_RESTRINGIDA\_POR\_DEFECTO."?err=1");

        }

    }

    else

    {

        echo "que paos";

        session\_destroy();

        die;

    }

}

else

{

    echo "Error de capa 8";

    $sesion = "";

}

?>

<?php

include "settings.php";

require\_once("funciones.php");

session\_start();

logout();

?>

Fuente: elaboración propia

Código de programación 5

*Archivo: settings.php*

<?php

// aquí se deberá copiar el hash MD5 obtenido con PHP CLI

// user: jorge

// pass: 1

define("HASH\_ACCESO","49a0b5b764cfce738da74da8e8e4222d");

// formulario.html será el que pida el ingreso de user y pass al usuario

//const PAGINA\_LOGIN = "formulario.html";

define("PAGINA\_LOGIN","index.php");

// esta será una página cualquiera, co n acceso restringido, a la cual rediri gir al usuario después de iniciar su sesión en el sistema

//const PAGINA\_RESTRINGIDA\_POR\_DEFECTO ="pagina\_de\_muestra";

define("PAGINA\_RESTRINGIDA\_POR\_DEFECTO","panel.php");

?>

Fuente: elaboración propia

Código de programación 6

*Archivo: index.php*

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="css/bootstrap.min.css">

    <link rel="stylesheet" href="css/bootstrap.css">

    <link rel="stylesheet" href="css/estilos.css">

    <title>Tarea semana 6</title>

</head>

    <body>

        <script src="js/jquery-3.6.0.min.js"></script>

        <script src="js/popper.min.js"></script>

        <script src="js/bootstrap.js"></script>

        <script src="js/funciones.js"></script>

        <div class="container-fluid">

            <div class="row">

                <div class="col-sm-12  bg-dark text-white text-center">

                    <h1>Login</h1>

                </div>

            </div>

            <div class="row bg-dark text-white">

                <div class="col-sm-12  bg-dark text-white">&nbsp; </div>

            </div>

            <div class="row">

                <div class="col-sm-0 col-md-1 col-xl-1  bg-dark text-white">

                </div>

                <div class="col-sm-12 col-md-10 col-xl-10  bg-dark text-center">

                    <form action="validar.php" method="post">

                            <img src="img/admin-icon.png" alt="Avatar" class="avatar">

                            <label for="usuario" class="text-white"><b>Usuario</b></label>

                            <input type="text" placeholder="Ingrese nombre de usuario" name="usuario" required>

                            <label for="psw" class="text-white"><b>Contraseña</b></label>

                            <input type="password" placeholder="Ingrese contraseña" name="psw" required>

                            <button type="submit" class="btn-outline-secondary">Iniciar sesion</button>

                    </form>

                    </div>

                <div class="col-sm-0 col-md-1 col-xl-1  bg-dark text-white"></div>

            </div>

            <div class="row bg-dark text-white">

                <div class="col-sm-12  bg-dark text-white">&nbsp; </div>

            </div>

            <div class="row bg-dark text-white">

                <div class="col-sm-12  bg-dark text-white">&nbsp; </div>

            </div>

            <div class="row bg-dark text-white">

                <div class="col-sm-12  bg-dark text-white">&nbsp; </div>

            </div>

        </div>

    </body>

</html>

Fuente: elaboración propia

Código de programación 7

*Archivo: panel.php*

<?php

include 'Configuracion.php';

session\_start();

$sw = false;

if(isset($\_GET['err']))

{

    if($\_GET['err'] == "1")

    {

        $sesion = "Error de usuario";

    }

}

else

{

    if(isset($\_SESSION['usuario']))

    {

        $sesion = $\_SESSION['usuario'];

        $sw = true;

        if(($sesion == null || $sesion == '') && !validar\_user\_y\_pass())

        {

            echo "No existe sesion actual";

            die();

        }

    }

    else

    {

        $sesion = "X";

    }

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="css/bootstrap.min.css">

    <link rel="stylesheet" href="css/bootstrap.css">

    <link rel="stylesheet" href="css/estilos.css">

    <title>Tarea semana 6</title>

</head>

    <body>

        <script src="js/jquery-3.6.0.min.js"></script>

        <script src="js/popper.min.js"></script>

        <script src="js/bootstrap.js"></script>

        <script src="js/funciones.js"></script>

        <div class="container-fluid">

            <div class="row">

                <div class="col-sm-12  bg-dark text-white text-center">

                    <h1>Carro de compras</h1>

                </div>

            </div>

            <div class="row bg-dark text-white">

                <div class="col-sm-12  bg-dark text-white">

                    <h2>

                        Benvenido <?php echo $sesion;?>

                        <a href="cerrarsesion.php" class="btn btn-outline-light pull-right">Cerrar sesion</a>

                    </h2>

                </div>

            </div>

            <div class="row bg-dark text-white">

                <div class="col-sm-12  bg-dark text-white">&nbsp; </div>

            </div>

            <div class="row bg-dark text-white">

                <div class="col-sm-12  bg-dark text-white">&nbsp; </div>

            </div>

        </div>

        <?php

        if ($sw)

        {

        ?>

        <div class="container">

            <div class="panel panel-default">

                <div class="panel-heading">

                    <ul class="nav nav-pills">

                        <li role="presentation" class="active"><a href="index.php">Inicio</a></li>

                        <li role="presentation"><a href="VerCarta.php">Ver carro de compras</a></li>

                    </ul>

                </div>

                <div class="panel-body">

                    <h1>Mis Productos</h1>

                    <br>

                    <br>

                    <a href="VerCarta.php" class="cart-link" title="Ver Carta"><i class="glyphicon glyphicon-shopping-cart pull-left"></i></a>

                    <br>

                    <br>

                    <br>

                    <div id="products" class="row list-group">

                        <?php

                        //get rows query

                        $query = $db->query("SELECT \* FROM mis\_productos ORDER BY id DESC LIMIT 10");

                        if($query->num\_rows > 0){

                            while($row = $query->fetch\_assoc()){

                        ?>

                        <div class="item col-lg-4">

                            <div class="row">

                                        <div class="col-md-12">

                                        </div>

                            </div>

                        </div>

                        <div class="item col-lg-4">

                            <div class="thumbnail">

                                <div class="caption">

                                    <h4 class="list-group-item-heading"><?php echo $row["name"]; ?></h4>

                                    <p class="list-group-item-text"><?php echo $row["description"]; ?></p>

                                    <div class="row">

                                        <div class="col-md-6">

                                            <p class="lead"><?php echo '$'.$row["price"].' Pesos'; ?></p>

                                        </div>

                                        <div class="col-md-6">

                                            <a class="btn btn-success" href="AccionCarta.php?action=addToCart&id=<?php echo $row["id"]; ?>">Agregar al carro</a>

                                        </div>

                                    </div>

                                </div>

                            </div>

                        </div>

                        <?php } }else{ ?>

                        <p>Producto(s) no existe.....</p>

                        <?php } ?>

                    </div>

                </div>

                <div class="panel-footer">Tarea semana 6</div>

            </div><!--Panek cierra-->

        </div>

        <?php

        }

        ?>

    </body>

</html>

Fuente: elaboración propia

Código de programación 8

*Archivo: verCarta.php*

<?php

include 'La-carta.php';

$cart = new Cart;

//session\_start();

$sw = false;

if(isset($\_SESSION['usuario']))

{

    $sesion = $\_SESSION['usuario'];

    $sw = true;

    if(($sesion == null || $sesion == '') && !validar\_user\_y\_pass())

    {

        echo "No existe sesion actual";

        die();

    }

}

else

{

    $sesion = "X";

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="css/bootstrap.min.css">

    <link rel="stylesheet" href="css/bootstrap.css">

    <link rel="stylesheet" href="css/estilos.css">

    <title>Tarea semana 6</title>

</head>

    <body>

        <script src="js/jquery-3.6.0.min.js"></script>

        <script src="js/popper.min.js"></script>

        <script src="js/bootstrap.js"></script>

        <script src="js/funciones.js"></script>

        <div class="container-fluid">

            <div class="row">

                <div class="col-sm-12  bg-dark text-white text-center">

                    <h1>Carro de compras</h1>

                </div>

            </div>

            <div class="row bg-dark text-white">

                <div class="col-sm-12  bg-dark text-white">

                    <h2>

                        Benvenido <?php echo $sesion;?>

                        <a href="cerrarsesion.php" class="btn btn-outline-light pull-right">Cerrar sesion</a>

                    </h2>

                </div>

            </div>

            <div class="row bg-dark text-white">

                <div class="col-sm-12  bg-dark text-white">&nbsp; </div>

            </div>

            <div class="row bg-dark text-white">

                <div class="col-sm-12  bg-dark text-white">&nbsp; </div>

            </div>

        </div>

        <?php

        if ($sw)

        {

        ?>

        <div class="container">

            <div class="panel panel-default">

                <div class="panel-heading">

                    <ul class="nav nav-pills ">

                        <li role="presentation" ><a href="panel.php">Inicio</a></li>

                        <li role="presentation" class="active"><a href="VerCarta.php">Ver carro de compras</a></li>

                    </ul>

                </div>

                <div class="panel-body">

                    <h1>Carrito de compras</h1>

                    <table class="table">

                    <thead>

                        <tr>

                            <th>Producto</th>

                            <th>Precio</th>

                            <th>Cantidad</th>

                            <th>Sub total</th>

                            <th>&nbsp;</th>

                        </tr>

                    </thead>

                    <tbody>

                        <?php

                        if($cart->total\_items() > 0){

                            //get cart items from session

                            $cartItems = $cart->contents();

                            foreach($cartItems as $item){

                        ?>

                        <tr>

                            <td><?php echo $item["name"]; ?></td>

                            <td><?php echo '$'.$item["price"].' pesos'; ?></td>

                            <td>

                                <input type="number" class="form-control text-center"

                                        value="<?php echo $item["qty"]; ?>"

                                        onchange="updateCartItem(this, '<?php echo $item['rowid']; ?>')">

                            </td>

                            <td><?php echo '$'.$item["subtotal"].' pesos'; ?></td>

                            <td>

                                <a href="AccionCarta.php?action=removeCartItem&id=<?php echo $item["rowid"]; ?>" class="btn btn-warning" onclick="return confirm('Confirma eliminar?')"><i class="glyphicon glyphicon-trash"></i></a>

                            </td>

                        </tr>

                        <?php } }else{ ?>

                        <tr><td colspan="5"><p>Tu carta esta vacia.....</p></td>

                        <?php } ?>

                    </tbody>

                    <tfoot>

                        <tr>

                            <td><a href="panel.php" class="btn btn-success"><i class="glyphicon glyphicon-menu-left"></i> Continue Comprando</a></td>

                            <td colspan="2"></td>

                            <?php if($cart->total\_items() > 0){ ?>

                            <td class="text-center"><strong>Total <?php echo '$'.$cart->total().' pesos'; ?></strong></td>

                            <td><a href="#" class="btn btn-success btn-block">Pagos <i class="glyphicon glyphicon-menu-right"></i></a></td>

                            <?php } ?>

                        </tr>

                    </tfoot>

                    </table>

                </div>

            <div class="panel-footer">Tarea semana 6</div>

            </div><!--Panek cierra-->

        </div>

        <?php

        }

        ?>

    </body>

</html>

Fuente: elaboración propia

Código de programación 9

*Archivo: La-carta.php*

<?php session\_start();

class Cart {

    protected $cart\_contents = array();

    public function \_\_construct(){

        // get the shopping cart array from the session

        $this->cart\_contents = !empty($\_SESSION['cart\_contents'])?$\_SESSION['cart\_contents']:NULL;

        if ($this->cart\_contents === NULL){

            // set some base values

            $this->cart\_contents = array('cart\_total' => 0, 'total\_items' => 0);

        }

    }

    /\*\*

     \* Cart Contents: Returns the entire cart array

     \* @param   bool

     \* @return  array

     \*/

    public function contents(){

        // rearrange the newest first

        $cart = array\_reverse($this->cart\_contents);

        // remove these so they don't create a problem when showing the cart table

        unset($cart['total\_items']);

        unset($cart['cart\_total']);

        return $cart;

    }

    /\*\*

     \* Get cart item: Returns a specific cart item details

     \* @param   string  $row\_id

     \* @return  array

     \*/

    public function get\_item($row\_id){

        return (in\_array($row\_id, array('total\_items', 'cart\_total'), TRUE) OR ! isset($this->cart\_contents[$row\_id]))

            ? FALSE

            : $this->cart\_contents[$row\_id];

    }

    /\*\*

     \* Total Items: Returns the total item count

     \* @return  int

     \*/

    public function total\_items(){

        return $this->cart\_contents['total\_items'];

    }

    /\*\*

     \* Cart Total: Returns the total price

     \* @return  int

     \*/

    public function total(){

        return $this->cart\_contents['cart\_total'];

    }

    /\*\*

     \* Insert items into the cart and save it to the session

     \* @param   array

     \* @return  bool

     \*/

    public function insert($item = array()){

        if(!is\_array($item) OR count($item) === 0){

            return FALSE;

        }else{

            if(!isset($item['id'], $item['name'], $item['price'], $item['qty'])){

                return FALSE;

            }else{

                /\*

                 \* Insert Item

                 \*/

                // prep the quantity

                $item['qty'] = (float) $item['qty'];

                if($item['qty'] == 0){

                    return FALSE;

                }

                // prep the price

                $item['price'] = (float) $item['price'];

                // create a unique identifier for the item being inserted into the cart

                $rowid = md5($item['id']);

                // get quantity if it's already there and add it on

                $old\_qty = isset($this->cart\_contents[$rowid]['qty']) ? (int) $this->cart\_contents[$rowid]['qty'] : 0;

                // re-create the entry with unique identifier and updated quantity

                $item['rowid'] = $rowid;

                $item['qty'] += $old\_qty;

                $this->cart\_contents[$rowid] = $item;

                // save Cart Item

                if($this->save\_cart()){

                    return isset($rowid) ? $rowid : TRUE;

                }else{

                    return FALSE;

                }

            }

        }

    }

    /\*\*

     \* Update the cart

     \* @param   array

     \* @return  bool

     \*/

    public function update($item = array()){

        if (!is\_array($item) OR count($item) === 0){

            return FALSE;

        }else{

            if (!isset($item['rowid'], $this->cart\_contents[$item['rowid']])){

                return FALSE;

            }else{

                // prep the quantity

                if(isset($item['qty'])){

                    $item['qty'] = (float) $item['qty'];

                    // remove the item from the cart, if quantity is zero

                    if ($item['qty'] == 0){

                        unset($this->cart\_contents[$item['rowid']]);

                        return TRUE;

                    }

                }

                // find updatable keys

                $keys = array\_intersect(array\_keys($this->cart\_contents[$item['rowid']]), array\_keys($item));

                // prep the price

                if(isset($item['price'])){

                    $item['price'] = (float) $item['price'];

                }

                // product id & name shouldn't be changed

                foreach(array\_diff($keys, array('id', 'name')) as $key){

                    $this->cart\_contents[$item['rowid']][$key] = $item[$key];

                }

                // save cart data

                $this->save\_cart();

                return TRUE;

            }

        }

    }

    /\*\*

     \* Save the cart array to the session

     \* @return  bool

     \*/

    protected function save\_cart(){

        $this->cart\_contents['total\_items'] = $this->cart\_contents['cart\_total'] = 0;

        foreach ($this->cart\_contents as $key => $val){

            // make sure the array contains the proper indexes

            if(!is\_array($val) OR !isset($val['price'], $val['qty'])){

                continue;

            }

            $this->cart\_contents['cart\_total'] += ($val['price'] \* $val['qty']);

            $this->cart\_contents['total\_items'] += $val['qty'];

            $this->cart\_contents[$key]['subtotal'] = ($this->cart\_contents[$key]['price'] \* $this->cart\_contents[$key]['qty']);

        }

        // if cart empty, delete it from the session

        if(count($this->cart\_contents) <= 2){

            unset($\_SESSION['cart\_contents']);

            return FALSE;

        }else{

            $\_SESSION['cart\_contents'] = $this->cart\_contents;

            return TRUE;

        }

    }

    /\*\*

     \* Remove Item: Removes an item from the cart

     \* @param   int

     \* @return  bool

     \*/

     public function remove($row\_id){

        // unset & save

        unset($this->cart\_contents[$row\_id]);

        $this->save\_cart();

        return TRUE;

     }

    /\*\*

     \* Destroy the cart: Empties the cart and destroy the session

     \* @return  void

     \*/

    public function destroy(){

        $this->cart\_contents = array('cart\_total' => 0, 'total\_items' => 0);

        unset($\_SESSION['cart\_contents']);

    }

}

Fuente: elaboración propia

Código de programación 10

*Archivo: funciones.js*

function updateCartItem(obj,id)

{

    $.get("AccionCarta.php", {action:"updateCartItem", id:id, qty:obj.value}, function(data)

    {

        if(data == 'ok')

        {

            location.reload();

        }

        else

        {

            alert('Cart update failed, please try again.');

        }

    });

}

Fuente: elaboración propia

Código de programación 11

*Archivo: accionCarta.php*

<?php

date\_default\_timezone\_set("America/Lima");

// Iniciamos la clase de la carta

include 'La-carta.php';

$cart = new Cart;

// include database configuration file

include 'Configuracion.php';

if(isset($\_REQUEST['action']) && !empty($\_REQUEST['action'])){

    if($\_REQUEST['action'] == 'addToCart' && !empty($\_REQUEST['id'])){

        $productID = $\_REQUEST['id'];

        // get product details

        $query = $db->query("SELECT \* FROM mis\_productos WHERE id = ".$productID);

        $row = $query->fetch\_assoc();

        $itemData = array(

            'id' => $row['id'],

            'name' => $row['name'],

            'price' => $row['price'],

            'qty' => 1

        );

        $insertItem = $cart->insert($itemData);

        $redirectLoc = $insertItem?'VerCarta.php':'index.php';

        header("Location: ".$redirectLoc);

    }elseif($\_REQUEST['action'] == 'updateCartItem' && !empty($\_REQUEST['id'])){

        $itemData = array(

            'rowid' => $\_REQUEST['id'],

            'qty' => $\_REQUEST['qty']

        );

        $updateItem = $cart->update($itemData);

        echo $updateItem?'ok':'err';die;

    }elseif($\_REQUEST['action'] == 'removeCartItem' && !empty($\_REQUEST['id'])){

        $deleteItem = $cart->remove($\_REQUEST['id']);

        header("Location: VerCarta.php");

    }elseif($\_REQUEST['action'] == 'placeOrder' && $cart->total\_items() > 0 && !empty($\_SESSION['sessCustomerID'])){

        // insert order details into database

        $insertOrder = $db->query("INSERT INTO orden (customer\_id, total\_price, created, modified) VALUES ('".$\_SESSION['sessCustomerID']."', '".$cart->total()."', '".date("Y-m-d H:i:s")."', '".date("Y-m-d H:i:s")."')");

        if($insertOrder){

            $orderID = $db->insert\_id;

            $sql = '';

            // get cart items

            $cartItems = $cart->contents();

            foreach($cartItems as $item){

                $sql .= "INSERT INTO orden\_articulos (order\_id, product\_id, quantity) VALUES ('".$orderID."', '".$item['id']."', '".$item['qty']."');";

            }

            // insert order items into database

            $insertOrderItems = $db->multi\_query($sql);

            if($insertOrderItems){

                $cart->destroy();

                header("Location: OrdenExito.php?id=$orderID");

            }else{

                header("Location: Pagos.php");

            }

        }else{

            header("Location: Pagos.php");

        }

    }else{

        header("Location: index.php");

    }

}else{

    header("Location: index.php");

}

Fuente: elaboración propia

Código de programación 12

*Archivo: estilos.css*

html

{

    margin: 0 auto;

    width: 95%;

}

a

{

    font-size: 200%;

}

.container

{

    padding: 20px;

}

.cart-link

{

    width: 100%;

    text-align: right;

    display: block;

    font-size: 22px;

}

input[type="number"]

{

    width: 20%;

}

Fuente: elaboración propia

Código de programación 13

*Archivo: mis\_productos.sql*

-- phpMyAdmin SQL Dump

-- version 5.1.0

-- https://www.phpmyadmin.net/

--

-- Servidor: 127.0.0.1

-- Tiempo de generación: 19-07-2021 a las 22:15:55

-- Versión del servidor: 10.4.18-MariaDB

-- Versión de PHP: 8.0.3

SET SQL\_MODE = "NO\_AUTO\_VALUE\_ON\_ZERO";

START TRANSACTION;

SET time\_zone = "+00:00";

/\*!40101 SET @OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET @OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET @OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/;

/\*!40101 SET NAMES utf8mb4 \*/;

--

-- Base de datos: `carta`

--

-- --------------------------------------------------------

--

-- Estructura de tabla para la tabla `mis\_productos`

--

CREATE TABLE `mis\_productos` (

  `id` int(11) NOT NULL,

  `name` varchar(200) COLLATE utf8\_unicode\_ci NOT NULL,

  `description` text COLLATE utf8\_unicode\_ci NOT NULL,

  `price` float(10,2) NOT NULL,

  `created` datetime NOT NULL,

  `modified` datetime NOT NULL,

  `status` enum('1','0') COLLATE utf8\_unicode\_ci NOT NULL DEFAULT '1'

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;

--

-- Volcado de datos para la tabla `mis\_productos`

--

INSERT INTO `mis\_productos` (`id`, `name`, `description`, `price`, `created`, `modified`, `status`) VALUES

(1, 'Juan Salvador Gaviota', 'Cuento', 5600.00, '2016-08-17 08:21:25', '2016-08-17 08:21:25', '1'),

(2, '2 años de vacaciones', 'Aventura', 10500.00, '2016-08-17 08:21:25', '2016-08-17 08:21:25', '1'),

(3, 'Necronomicon', 'Terror', 23600.00, '2016-08-17 08:21:25', '2016-08-17 08:21:25', '1'),

(4, 'La biblia de los caidos: El testamento de MAD', 'Ficcion', 15500.00, '2016-08-17 08:21:25', '2016-08-17 08:21:25', '1'),

(5, 'La guerra de los cielos', 'Ficcion', 14500.00, '2021-07-19 09:33:58', '2021-07-19 09:33:58', '1');

--

-- Índices para tablas volcadas

--

--

-- Indices de la tabla `mis\_productos`

--

ALTER TABLE `mis\_productos`

  ADD PRIMARY KEY (`id`);

--

-- AUTO\_INCREMENT de las tablas volcadas

--

--

-- AUTO\_INCREMENT de la tabla `mis\_productos`

--

ALTER TABLE `mis\_productos`

  MODIFY `id` int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=6;

COMMIT;

/\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION \*/;

Fuente: elaboración propia

***Vistas en navegador***

Figura 1

*Vista de index.php en navegador*

Interfaz de usuario gráfica, Aplicación, Teams

Descripción generada automáticamente

Fuente: elaboración propia

Figura 2

*Vista de panel.php en navegador cuando el usuario es incorrecto.*

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Fuente: elaboración propia

Figura 3

*Vista de panel.php en navegador cuando el usuario es correcto.*

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Fuente: elaboración propia

Figura 4

*Vista de VerCarta.php en navegador*

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Fuente: elaboración propia

Bibliografía

IACC. (2020). Sesiones y Control de Usuarios con PHP. En *Programación avanzada 1. Semana 6.*

IACC. (2021). Tarea semana 6.

Tapia, N. (2018). *BaulPHP*. Obtenido de https://www.baulphp.com/carrito-de-compras-con-php-y-mysql/

The PHP Group. (s.f.). *Manual de PHP*. Obtenido de https://www.php.net/manual/es/index.php

*w3schools.com*. (s.f.). Obtenido de https://www.w3schools.com/