

**APLICACIONES EN INTERNET**  
**EXAMEN DE PROBLEMAS- FEBRERO 2020**

1. Una tienda de comida rápida utiliza una aplicación para la gestión de pedidos a domicilio. La aplicación usa un servidor MySQL con una BBDD llamada *pedidos* en donde se almacena, entre otras, la tabla: *usuarios* cuyos campos son *id*, *nombre*, *direccion*, con identificador único, nombre y dirección del usuario. Además, utiliza una tabla para los menús que sirve, llamada *menus*, con campos *id*, *descripcion*, *precio*. Los pedidos que realiza un usuario se almacenan en una tabla *pediosusuarios* cuyos campos son *id*, identificador único de los pedidos, *id\_u* identificador de usuario, *id\_p*, identificador de menú, y *date*, la fecha en que se realizó.

**No se limite a escribir código, describa brevemente su solución.**

- a) Realice una función PHP que reciba un *id* de usuario, genere un formulario HTML para realizar un pedido. Para ello, en el formulario se mostrará el nombre del usuario y se proporcionará una lista de selección con el nombre y precio de todos los menús de la tienda. El usuario solo puede seleccionar un menú en este paso **(0.75 puntos)**.

```
function generarFormulario($id) {  
    //Conectar a BBDD: $pdo  
    $q="SELECT nombre FROM usuarios WHERE id=".$id;  
    $r=$pdo->query($q);  
    $n=$r->fetch(FETCH_ASSOC);  
    echo '<form action="sel.php" method=GET">';  
    echo '<input type="hidden" name="u_id" value="'.$id.'>';  
    echo " Usuario: ".$n['nombre'];  
    echo '<select name="menu">';  
    $q="SELECT * FROM menus";  
    $r=$pdo->query($q);  
    while($l=$r->fetch(FETCH_ASSOC)) {  
        echo '<option value="'.$l['id'].'">'.$l['descripcion'].'</option>';  
    }  
    echo '</select></form>';  
    //Se podria Tambien añadir un botón de enviar  
}
```

- b) Proporcione un script PHP que procese el formulario del apartado anterior y, como siguiente paso, permita al usuario seleccionar otro menú para su pedido en curso. Este script mostrará de nuevo el formulario anterior y un botón para terminar pedido. Para mantener la información entre pasos debe **usar cookies (1 punto)**.

```
//sel.php  
setcookie($_GET['menu'],1);  
setcookie("usuario",$_GET['id_u']);  
generarFormulario($_GET['id_u']);  
echo '<form action="terminar.php" method="POST"><input type="submit"></input></form>';
```

- c) Realice un script en PHP que procese el botón de terminar pedido del apartado anterior. Es decir, inserte un pedido en la BBDD, con la información de los pasos anteriores. Puede suponer que existe ya implementada una función que genera un identificador único para el pedido **(1 punto)**.

```
//terminar.php  
//usamos las cookies recibidas  
//Conectar a BBDD: $pdo  
$idp=generarIdUnico(); //Genera el id de pedido  
$idu=$_COOKIE['usuario'];  
$fecha=time();
```

```

foreach ($_COOKIE as $k=>$v) {
    if ($k!="usuario") {
        $q="INSERT INTO pedidosusuarios VALUES('.$idp.','.$ipu.','.$v.','.$fecha.')";
        $pdo->query($q);
    }
}

```

- d) Se pretende mejorar la forma en la que se seleccionan los menús en los pedidos (apartado a) anterior, de manera que se puedan seleccionar varios menús en el mismo paso, utilizando Javascript. Para ello se mostrará una tabla HTML con todas los menús y precios, como en el siguiente fragmento de código HTML (numeración incluida para facilitar la respuesta si es necesario):

```

1. Menús seleccionados: <ol></ol>
2. <table>
3. <tr><td>1</td><td>Hamburguesa</td>10.5</tr>
4. <tr><td>2</td><td>Bravas</td>4.5</tr>
5. ... NO SE MUESTRA EL RESTO DE ENTRADAS
6. </table>

```

Para seleccionar los menús del pedido el usuario pinchará en las filas de la tabla de menús. Es decir, cada vez que el usuario pinche en una fila se añadirá el nombre del menú a la lista. Para ello, mediante **JavaScript exclusivamente** haga que cada vez que un usuario pincha en cualquiera de las filas se añada el nombre del menú y su precio a la lista de menús seleccionados. La lista se mostrará insertando un nuevo elemento en la lista de numeración que ya existe. Asocie adecuadamente dicho código Javascript al fragmento HTML (**1.25 punto**).

Para asociar, insertamos el siguiente código **al final del código HTML**

```

<script src="código.js" type="text/javascript"> asignarEventos();</script>
function seleccionar(e) {
    var ol=document.getElementsByTagName("OL")[0];
    var tr=e.target;
    var nombre=tr.children[1];
    var precio=tr.children[2];
    var li=document.CreateElement("LI");
    li.innerHTML=nombre+"\t"+precio;
    ol.appendChild(li);
}
function asignarEventos() {
    var trs=document.getElementsByTagName("TR");
    for (var i=0; i<trs.length; i++){
        trs[i].onclick=seleccionar;
    }
}

```

- e) Finalmente, modifique el apartado anterior o proporcione nuevo código que permita que el usuario borre menús de los seleccionados. Para ello, cuando el usuario pinche en alguno de los menús que ya se ha seleccionado, se elimine de la lista de selección (**1 punto**).

```

//Se declara la funcion borrar
function borrar(e) {
    e.target.remove();
}

```

//Se puede modificar directamente la función seleccionar para añadir el evento al elemento li credo

//al final de seleccionar  
li.onclick=borrar;

2. Consideremos una aplicación de descarga de películas que incorpora un sistema de recomendación por filtrado colaborativo para el que se definen  $n = 3$  características. Contamos con la siguiente base de datos de puntuaciones (definidas entre 0 y 5):

	Andrés	María	Carlos	Eusebio	Laura
Película 1	4	4	?	0	?
Película 2	3	?	2	?	4
Película 3	?	4	4	4	?
Película 4	?	3	4	?	1

Se dispone en MATLAB de las matrices:  $Y$  es la matriz de puntuaciones, **en la que cada entrada no definida contiene "-1"**, y la matriz  $x$  contiene las características de las películas, para  $n$  características distintas, y  $\Theta$  los parámetros de los usuarios:

$$X = \begin{bmatrix} \text{---} (x^{(1)})^T \text{---} \\ \text{---} (x^{(2)})^T \text{---} \\ \vdots \\ \text{---} (x^{(n_m)})^T \text{---} \end{bmatrix}, \quad \Theta = \begin{bmatrix} \text{---} (\theta^{(1)})^T \text{---} \\ \text{---} (\theta^{(2)})^T \text{---} \\ \vdots \\ \text{---} (\theta^{(n_u)})^T \text{---} \end{bmatrix}$$

siendo  $n_m$  es el número de películas y  $n_u$  el número de usuarios

- a) Indique las dimensiones de las distintas matrices usadas y el código MATLAB para obtener una matriz  $R$  en la que,  $R(i,j)$  es 1 si el usuario  $i$  ha puntuado la película  $j$  y 0 en caso contrario (**1 punto**).

$Y: [n_m, n_u]$ .  $\Theta: [n_u, n]$ .  $X: [n_m, 3]$ .  $R=Y$ ,  $R(R>0)=1$ ;  $R(R<0)=0$ ;

- b) Escriba el código MATLAB necesario para calcular la función de coste  $J$  regularizada (**1 punto**).

$\text{Err} = (X * \Theta' - Y) .* R$ ;

$J = \text{sum}(\text{Err}(:).^2)/2 + \text{lambda} * (\text{sum}(X(:).^2) + \text{sum}(\Theta(:).^2))/2$ ;

- c) Suponga que tras ejecutar el algoritmo, se obtienen los siguientes valores para  $x$  y  $\Theta$ :

$X = [0.2 \ 0.5 \ 1; \ 0 \ 1 \ 1; \ 1 \ 0 \ 0.5; \ 0 \ 0.5 \ 1]$ ;

$\Theta = [1 \ 0 \ 3; \ 2 \ 1 \ 0; \ 3 \ 2 \ 0; \ 4 \ 1 \ 1; \ 0 \ 2 \ 1]$ ;

Indique qué puntuación se predice para el usuario Andrés y para la película 3. Indique cómo comprobar si el resultado ha sido fiable y aplíquelo al usuario Andrés. Justifique sus respuestas (**1.5 punto**).

La predicción corresponde al producto del vector de características de la película, por el de parámetros de usuario:  $x=X(3,:)= [1 \ 0 \ 0.5]$  y  $t=\Theta(:,1)= [1 \ 0 \ 3]$ ;  $yp=x*t' = 2.5$ .

En general, para comprobar la calidad de nuestra predicción tenemos que calcular el error cuadrático medio entre la predicción que proporcionan nuestros vectores de parámetros y características y las puntuaciones reales otorgadas por los usuarios. Para el caso de Andrés, podemos comprobar que  $e1=(yp1-yr1).^2=(x1*t'-yr).^2=(3.2-4).^2=0.64$ ,  $e2=(3-3)=0$ , por lo que tendríamos un error cuadrático medio de  $(0.64+0)/2=0.32$ .

- d) Dados los valores de  $X$  y  $\Theta$  del apartado anterior, suponga que quiere recomendar al usuario Andrés una película que haya visto ya un usuario con gustos similares. Indique justificadamente si usaría a María o a Carlos, y si le recomendaría la película 3 **(1.5 puntos)**.

Un usuario con gustos similares sería aquel que tiene un vector de parámetros “cercanos” al de Andrés. La distancia euclídea (se puede usar también la distancia euclídea al cuadrado para evitar la raíz cuadrada, es válido), entre Andrés y María es  $d1 = |\Theta(:,1) - \Theta(:,2)| = 3.316$  y con Carlos es  $d2 = 4.123$ . Por tanto, María tendría gustos más parecidos a Andrés y usaríamos a María. Para recomendar la película 3 tenemos que la predicción para esa película de Andrés es de 2.5 (apartado anterior), lo que no nos aclara demasiado. La predicción de Andrés para la película 4 es de 3 y podemos comprar la “similaridad” entre ambas películas y una bien puntuada por Andrés, por ejemplo la 4, y tenemos que  $|X(:,1) - X(:,3)| = 1.06$  y  $|X(:,1) - X(:,4)| = 0.2$ . Por tanto, la 4 también es más similar a la bien puntuada por Andrés, con lo que recomendamos la 4.