



# Benemérita Universidad Autónoma de Puebla

## Práctica #6. PHP y MySQL

Aplicaciones web

Facultad de Ciencias de la Computación

Equipo:

Vargas Arenas Pedro - 201734553



## Descripción

En este documento se describe el desarrollo de la sexta práctica de la materia que consiste en elaborar un programa que muestre un panel donde se desplieguen los nombres de los votantes disponibles, el usuario elegirá un nombre y votará por uno de los cuatro partidos políticos visualizados, o bien, podrá anular su voto. El nombre y el voto de todos los usuarios que hayan votado serán almacenados en un base de datos, así como la información de cada partido político.

Se visualizará en una tabla, en un PDF y en la pantalla tanto la cantidad de votos por partido político como los votos anulados.

## Desarrollo

### Panel de votos

Este módulo se desarrolló anidando un formulario y una tabla. El formulario es el elemento principal, ya que posee los elementos para seleccionar el nombre, el voto y el botón para enviar la información. Para esto, se realizó una conexión a la base de datos y una consulta para obtener los nombres de los votantes.

```
$link = mysqli_connect("localhost", "root", "")
    or die ("Error al conectar con base de datos");
mysqli_set_charset($link, "UTF8");
mysqli_select_db($link, "elecciones");
$result = mysqli_query($link, "SELECT * FROM votante");
```

Para mostrar los nombres, se utiliza un ciclo while que recorrerá cada fila obtenida de la consulta.

```
<form action="votos.php" method="POST" style="text-align:center;">
    Nombre del votante:
    <select name="nombre">
        <?php
            while ($row = mysqli_fetch_array($result)) {
                $nombre = $row['nombre'];
                echo "<option>$nombre</option>";
            }
        ?>
    </select>
```

Los elementos son presentados en una tabla, la primera fila se compone de los logos de los partidos políticos y del ícono para anular el voto.

```
<tr>
    <th></th>
    <th></th>
    <th></th>
    <th></th>
    <th></th>
</tr>
```

La segunda fila contiene los nombres de cada partido y la leyenda “anular voto”.

```
<tr>
    <td>PRI </td>
    <td>PAN </td>
    <td>PRD </td>
    <td>MORENA </td>
    <td>ANULAR VOTO </td>
</tr>
```

La tercera fila posee los elementos *input radio* para capturar el voto del usuario, el valor asignado es el nombre del partido político por el cual votó.

```
<tr>
    <td><input type="radio" name="voto" value="1"></td>
    <td><input type="radio" name="voto" value="2"></td>
    <td><input type="radio" name="voto" value="3"></td>
    <td><input type="radio" name="voto" value="4"></td>
    <td><input type="radio" name="voto" value="5"></td>
</tr>
```

Finalmente, se coloca el botón para enviar la información ingresada al archivo *votos.php* mediante el método post.

```
<p><input type="submit" value="Enviar datos"></p>
```

## Construcción de la base de datos

La base de datos desarrollada se llama *elecciones* que cuenta con dos tablas, la primera tabla es *partido* que cuenta con los siguientes campos:

- id\_partido: Valor numérico para asignar un ID a cada partido registrado.
- nombre: Contiene el nombre del partido.
- imagen: Posee el nombre de la imagen del partido.
- descripcion: Asigna una pequeña descripción de cada partido.

La segunda tabla se llama *votante* que posee los siguientes campos:

- INE: Contiene el ID asignado del votante.
- nombre: Posee el nombre del votante.
- edad: Valor numérico que contiene la edad del individuo.
- direccion: Contiene la dirección del votante.
- fecha\_nacimiento: Posee la fecha de nacimiento del votante seleccionado.

Por último, la tabla *voto* contendrá los votos realizados, posee los siguientes campos:

- id\_voto: Es el ID que se asignará a cada voto realizado.
- INE: Posee el número de INE del votante que ejerció el voto.
- voto: Contiene el ID del partido político por el que se votó.

## Manejo de la información

Una vez que se recibe la información con ayuda de la propiedad `$_REQUEST[""]` en las variables `$nombre` y `$voto` del archivo *votos.php*, se conecta la base de datos mediante las funciones de `mysqli` y se realiza la consulta `"SELECT * FROM voto"` para obtener todos los votos, esta es almacenada en una variable.

```
$nombre = $_REQUEST['nombre'];
$voto = $_REQUEST['voto'];
$nVotos = [0, 0, 0, 0, 0];

$link = mysqli_connect("localhost", "root", "")
    or die ("Error al conectar con base de datos");
mysqli_set_charset($link, "UTF8");
mysqli_select_db($link, "elecciones");
$result = mysqli_query($link, "SELECT * FROM voto");
```

El voto y la información recibida se debe almacenar en la tabla *votantes*, para ello, se utiliza la función `insertarVoto($link, $nombre, $voto)`. Dentro de la función se busca el INE del votante el cual se almacenado en la tabla *voto*. Para esto, se realiza la operación `INSERT`.

```
function insertarVoto($link, $nombre, $voto) {
    $votante = mysqli_query($link, "SELECT INE FROM votante WHERE nombre = '$nombre'")
        or die ("Error al buscar votante");
    while ($row = mysqli_fetch_array($votante))
        $id = $row['INE'];
    $res = mysqli_query($link, "INSERT INTO voto (INE, voto)
        VALUES ('$id', '$voto')")
        or die ("Error al insertar el voto");
}
```

Los votos almacenados en la base de datos se deben obtener de la consulta para guárdalos en un arreglo. Esto se ejecuta dentro de la función *obtenerVotos(\$link, \$nVotos, \$result)* mediante un ciclo while que se ejecuta mientras existan elementos dentro de la consulta. Solo se obtienen los datos de la columna voto para ser agregados al correspondiente lugar del arreglo *\$nVotos*.

```
function obtenerVotos($link, $nVotos, $result){  
    while ($row = mysqli_fetch_array($result)) {  
        $id = $row['voto'];  
        $nVotos = sumarVoto($id, $nVotos);  
    }  
    return $nVotos;  
}
```

La función *sumarVoto(\$voto, \$nVotos)* compara el voto realizado por partido para sumarlo al acumulado dentro del arreglo.

```
function sumarVoto($voto, $nVotos) {  
    switch ($voto) {  
        case 1:  
            $nVotos[0] += 1;  
            break;  
        case 2:  
            $nVotos[1] += 1;  
            break;  
        case 3:  
            $nVotos[2] += 1;  
            break;  
        case 4:  
            $nVotos[3] += 1;  
            break;  
        case 5:  
            $nVotos[4] += 1;  
            break;  
        default:  
            break;  
    }  
    return $nVotos;  
}
```

## Visualización de la información

Por último, se debe mostrar el total de votos por partido en la página, también se debe crear una tabla con la información de los votos de cada partido y un PDF con la misma información.

Primero se despliega la información en la página con ayuda de la función *mostrarVotos(\$nVotos)* que recibe el arreglo de votos. Esto se muestra junto con una cadena con etiquetas de HTML. Además, muestra dos enlaces para acceder al PDF y la gráfica creadas previamente.

```
function mostrarVotos($nVotos, $nombre, $voto) {

    $nombres = ["PRI", "PAN", "PRD", "MORENA", "VOTO ANULADO"];
    $cadenaVoto = "votos";
    $partido = buscarPartido($nombres, $voto);
    echo "<h3 style='color:navy;'>$nombre votó por $partido.</h3>";
    echo "<br><br>";
    for($i = 0; $i < 5; $i++){
        if($nVotos[$i] == 1)
            $cadenaVoto = "voto";
        echo "<h3><font face='\".roman.\"'>\".$nombres[$i].\": \".$nVotos[$i].\" \".$cadenaVoto.\"</font></h3>";
        $cadenaVoto = "votos";
    }
    echo "<ul><li><a target='_blank' href='generated/votos.jpg'>Ver gráfica</a></li>";
    echo "<li><a target='_blank' href='votos.pdf'>Ver PDF</a></li></ul>";
}
```

Para realizar la gráfica se debe requerir la librería *libchart*. Primero se realiza una consulta para obtener el nombre de los partidos, después se crea la gráfica mediante la clase *HorizontalBarChart()* y se agrega cada nombre con la cantidad de votos correspondiente. Finalmente, se agrega toda la información y se realizan configuraciones para crear dicha gráfica. Esto se realiza en la función *grafica(\$link, \$nVotos)*.

```
function grafica($link, $nVotos) {

    $res = mysqli_query($link, "SELECT nombre FROM partido");
    $n = 0;
    $chart = new HorizontalBarChart(600, 270);
    $data = new XYDataSet();

    while ($row = mysqli_fetch_array($res)) {
        $nombre = $row['nombre'];
        $data->addPoint(new Point("$nombre", "$nVotos[$n]"));
        $n++;
    }
    $chart->setDataSet($data);
    $chart->getPlot()->setGraphPadding(new Padding(5, 30, 20, 240));
    $chart->setTitle("Votos por partido");
    if (file_exists("generated/votos.jpg"))
        unlink("generated/votos.jpg");
    $chart->render("generated/votos.jpg");
}
```

Por último, para crear el PDF se utiliza la librería *fpdf16*. De igual manera, se realiza una consulta a la base de datos que obtendrá el nombre de los partidos políticos. Posteriormente, se realizan diversas configuraciones para ajustar el formato que tendrá el contenido. Finalmente, se anexa el nombre del partido junto con la cantidad de votos para la posterior creación del archivo. Esto es realizado en la función *crearPDF(\$nVotos, \$link)*.

```
function crearPDF($nVotos, $link) {

    $pdf = new FPDF();
    $res = mysqli_query($link, "SELECT nombre FROM partido");
    $n = 0;

    $pdf->AddPage();
    $pdf->SetFont('Arial', 'B', 16);
    $pdf->Cell(150, 40, 'Cantidad de votos', 0, 1);
    $pdf->SetFont('Arial', 'B', 14);
    $pdf->Cell(0, 10, 'Partido    Votos', 0, 1);
    $pdf->Cell(0, 10, '_____ ', 0, 1);
    $pdf->SetFont('Arial', '', 10);
    while($row = mysqli_fetch_array($res)) {
        $id = $row["nombre"];
        $pdf->Cell(0, 10, $id.' --- '.$nVotos[$n], 0, 1);
        $n++;
    }
    $pdf->SetTextColor(0, 0, 255);
    $pdf->SetFont('', 'U');
    if (file_exists("votos.pdf"))
        unlink("votos.pdf");
    $pdf->Output("votos.pdf");
}
```

## Ejecución

Primeramente, se debe activar WampServer para activar un servidor y así lograr ejecutar PHP en el navegador. Posteriormente, se ejecuta el archivo index.php, ya que contiene el panel para votar.

## Panel de votación

Nombre del votante: Leonardo Zepeda Pérez 

				
PRI	PAN	PRD	MORENA	ANULAR VOTO
<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Enviar datos

En este caso, el usuario Leonardo Zepeda Pérez votará por el PAN. Una vez enviada la información, se sumará el voto con el resto en la base de datos. A continuación, se muestra las tres formas de visualización de los votos.

**Leonardo Zepeda Pérez votó por PRD.**

**PRI: 0 votos**

**PAN: 2 votos**

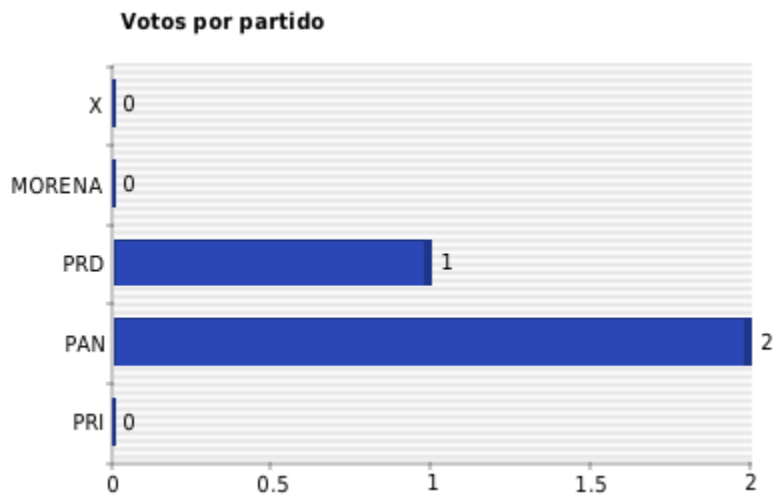
**PRD: 1 voto**

**MORENA: 0 votos**

**VOTO ANULADO: 0 votos**

Visualización de la página





Visualización de la gráfica

## Cantidad de votos

### Partido Votos

---

PRI --- 0

PAN --- 2

PRD --- 1

MORENA --- 0

X --- 0

Visualización del PDF

## Referencias

<https://developer.mozilla.org/es/docs/Learn/HTML/Tables/Basics>

[https://developer.mozilla.org/es/docs/Learn/Forms/Your first form](https://developer.mozilla.org/es/docs/Learn/Forms/Your_first_form)

[https://www.aprenderaprogramar.com/index.php?option=com\\_content&view=article&id=612:php-consultas-mysql-mysqlconnect-selectdb-query-fetcharray-freeresult-close-ejemplos-cu00841b&catid=70&Itemid=193](https://www.aprenderaprogramar.com/index.php?option=com_content&view=article&id=612:php-consultas-mysql-mysqlconnect-selectdb-query-fetcharray-freeresult-close-ejemplos-cu00841b&catid=70&Itemid=193)

<https://www.raulprietofernandez.net/blog/programacion/como-conectar-php-con-una-base-de-datos-mysql>