

Aplicatie

In care creem un formular pentru sondaje in PHP si prelucram si afisam rezultatele voturilor.

Formularul pe care il vom defini este unul simplu ce contine o intrebare si doua variante de raspuns: DA/NU. Pentru inceput, vom crea doua fisiere : voteaza.php - ce va afisa sondajul cu variantele de raspuns si va prelucra voturile si rezultate.php - ce va calcula numarul de voturi obtinute de fiecare raspuns.

Fisierul voteaza.php

In acest fisier vom verifica mai intai daca avem sau nu date transmise prin apasarea butonului de submit din formular.

Formularul pe care il vom folosi

si are urmatoarea forma:

```
<form action="<?php echo $PHP_SELF; ?>" method="POST">

    <div class="form">

        <?php echo "$intrebare"; ?>

    </div>

    <input type="radio" name="vote" value="da"/>Da<br/>

    <input type="radio" name="vote" value="nu"/>Nu<br/>

    <br/>

    <input type="submit" name="submit" value="voteaza"
class="btn"/>

</form>
```

Dupa cum poti observa, atributul action al tag-ului form este initializat cu variabila \$PHP_SELF, ceea ce semnifica fisierul .php curent (cel care este afisat in bara de navigare a browserului). In locul intrebării am afisat variabila \$intrebare, ce va trebui initializata inainte de

afisarea formularului cu un text ce poate fi citit fie dintr-o baza de date, fie dintr-un fisier. Pentru acest exemplu, eu am initializat variabila

astfel:

```
$intrebare = "E util acest tutorial?"
```

Ce mai poti observa este ca intrebarea este urmata de doua campuri input de tip radio-button, ce reprezinta cele doua variante de raspuns, si un buton de submit.

Pentru a verifica daca un utilizator de la o anumita adresa IP a votat deja, vom seta un cookie ce expira in 24 de ore. Cu alte cuvinte, un utilizator nu va putea vota decat o singura data pe zi.

OBS. Un cookie este un fisier text ce este salvat de catre browser pe calculatorul utilizatorului. Un cookie retine una sau mai multe perechi de tipul nume-valoare, ce pot fi encriptate pentru a securiza datele.

Cookie-ul va fi initializat utilizand functia predefinita **setcookie()** din php, ce are ca parametri numele cookie-ului, valoarea si ora cand expira. Timpul de expirare este calculat cu functia **time()** – ce returneaza data si ora curenta exprimate in secunde, la care adaugam o durata de 24 de ore, exprimata de asemenea in secunde.

```
<?php
// memoram adresa ip a utilizatorului in variabila $ip
$ip = $_SERVER['REMOTE_ADDR'];

    // calculam timpul de expirare al cookie-ului in secunde
    (timpul curent + 24 h)
$expira = time()+60*60*24;

    // setam cookie-ul
setcookie("user", "$ip", $expira);

?>
```

OBS. In codul de mai sus poti observa ca adresa IP a utilizatorului poate fi preluata din variabila superglobala **\$_SERVER, mai precis ea este data de proprietatea **\$_SERVER['REMOTE_ADDR']**.**

Pentru a verifica daca de la o adresa s-a votat deja vom utiliza variabila `$_COOKIE['nume cookie']`, in cazul nostru `$_COOKIE['user']`.

OBS. Cookie-urile nu reprezinta o metoda foarte sigura de a te asigura ca un utilizator voteaza o singura data in 24h. Ele pot fi sterse din browser, ceea ce inseamna ca utilizatorul va putea accesa din nou formularul. Daca vrei sa limitezi accesul la anumite zone importante dintr-un site, nu te baza pe cookie-uri, ci utilizeaza formulare de login si o baza de date MySQL.

Iata cum va arata codul final din fisierul `voteaza.php`:

```
<?php

// Verificam daca am primit date prin POST (daca butonul de
submit este setat)

if(!isset($_POST['submit'])) {

    // Daca nu avem date transmise

    // Verificam daca exista un cookie 'user' (daca s-a
votat deja de la ip-ul respectiv)

    if(isset($_COOKIE['user'])) {

        // Daca s-a votat deja, include fisierul ce
calculeaza rezultatele

        include "rezultate.php";

    } else {

        // Seteaza textul intrebarii si afiseaza
formularul

        $intrebare = "E util acest tutorial?";

    }

    <form action="<?php echo $PHP_SELF; ?>" method="post">

        <div class="form">

            <?php echo "$intrebare"; ?>

        </div>

        <input type="radio" name="vote" value="da"/>Da<br/>
```

```

        <input type="radio" name="vote" value="nu"/>Nu<br/>
        <br/>
        <input type="submit" name="submit" value="voteaza"
class="btn"/>
    </form>

    <?
    }
} else {
    // Daca avem date transmise
    // Verificam daca exista vectorul de raspunsuri
    if($_POST['vote']){
        // memoram adresa ip a utilizatorului in
variabila $ip
        $ip = $_SERVER['REMOTE_ADDR'];
        // calculam timpul de expirare al cookie-ului in
secunde (timpul curent + 24 h)
        $expira = time()+60*60*24;
        // setam cookie-ul
        setcookie("user", "$ip", $expira);
        // deschide fisierul text pentru inregistrarea
voturilor
        define(LOG_FILE, "voteaza.log");
        $f = @fopen(LOG_FILE, 'a+');
        if ($f) {
            // scriem in fisierul log ora, ip-ul utilizatorului
si raspunsul selectat
            @fputs($f, date("m.d.Y
g:ia").$_SERVER['REMOTE_ADDR']." ".$_POST['vote']."\n");

```

```

        @fclose($f);
    }

    // includem fisierul .php care afiseaza
    rezultatele

    include "rezultate.php";

    } else{

        // daca nu avem raspunsuri in POST, afisam un
        mesaj si reincarcam pagina

        echo "Nu ai ales nici o varianta.";

        echo "<meta http-equiv='refresh' content='1'>";

    }
}

?>

```

Dupa cum poti observa in codul de mai sus, pe langa lucrurile discutate deja am introdus si scrierea raspunsurilor utilizatorilor intr-un fisier text ("voteaza.log"), deoarece pentru a calcula numarul de raspunsuri DA sau NU, avem nevoie de un istoric al tuturor voturilor. Desigur, in locul fisierului text poti utiliza si o tabela MySQL.

In plus, pentru a afisa procentajul din fiecare raspuns, am creat un fisier rezultate.php ce este incarcat in voteaza.php cu instructiunea include:

```
include "rezultate.php"
```

Fisierul rezultate.php

Acest fisier va citi in primul rand toate inregistrarile din fisierul de log si le va retine in variabila \$continut:

```

<?php

$ fisier = "voteaza.log";

$ fh = fopen($ fisier, 'r');

$ continut = fread($ fh, filesize($ fisier));

fclose($ fh);

```

?>

Pentru a calcula numarul de voturi Da si Nu, vom folosi functia predefinita `str_replace()`. Aceasta functie este utilizata in special pentru a inlocui o anumita secventa sau sablon dintr-un text cu o alta secventa de caractere, dar pentru acest exemplu am utilizat-o pentru a numara aparitiile textelor „da” si „nu”. Ea are ca parametri: sablonul sau sirul de caractere inlocuit (da/nu), sirul de caractere ce inlocuieste sablonul (pe care il lasam gol), sirul de caractere in care cautam (in acest caz continutul fisierului) si numarul de aparitii ale sablonului.

Iata codul ce va calcula numarul de raspunsuri:

```
<?php

// Calculeaza numarul de voturi "Da"

$raspuns1 = "da";

$str = str_replace("$raspuns1", "", "$continut", $count1);

// Calculeaza numarul de voturi "Nu"

$raspuns2 = "nu";

$str = str_replace("$raspuns2", "", "$continut", $count2);

Intr-un final, fisierul nostru va arata astfel:

<?php

// Deschide fisierul text si memoreaza continutul in variabila
$continut

$intrebare = "E util acest tutorial?";

$fișier = "voteaza.log";

$fh = fopen($fișier, 'r');

$continut = fread($fh, filesize($fișier));

fclose($fh);

// Calculeaza numarul de voturi "Da"

$raspuns1 = "da";

$str = str_replace("$raspuns1", "", "$continut", $count1);
```

```

// Calculeaza numarul de voturi "Nu"
$raspuns2 = "nu";
$str = str_replace("$raspuns2", "", "$continut", $count2);
// Calculeaza numarul total de voturi
$total = $count1 + $count2;
// Afiseaza intrebarea
echo "<div class='intrebare'>";
echo "$intrebare";
echo "</div>";
// Rezultat imagine voturi "Da"
$raspuns1 = $count1/$total*140;
echo "Da ($count1 voturi)<br/>
    <div class='linie'>
        <div style='width: ".$raspuns1."px; height:12px;
background-color:#E20000;'></div>
    </div>";
// Rezultat imagine voturi "Nu"
$raspuns2 = $count2/$total*140;
echo "Nu ($count2 voturi)<br/>
    <div class='linie'>
        <div style='width: ".$raspuns2."px; height:12px;
background-color:#EB0000;'></div>
    </div>";
echo "<br/>Total: $total voturi<br/>";
?>

```

Dupa cum poti observa, dupa ce am calculat numarul de voturi, le-am afisat intr-o forma grafica utilizand div-uri si formatare css.