

## **Utilizarea limbajului PHP în aplicații multimedia. Crearea și manipularea imaginilor**

**Asistent Maria Radu**  
**Universitatea “Tibiscus” din Timișoara**

**ABSTRACT:** The purpose of a Web site is to interact with your users, whenever they might be. An extremely effective way of interacting with users is through the use of visuals. PHP offers an interface to the GD module, which in turn currently supports the creation and management of JPEGs and PNGs. By using the functions provided, you can create, add text to, draw pictures on, and modify preexisting PNG and JPEG images.

### **1 Introducere**

O aplicație multimedia este o combinație între grafică și program menită să creeze o interfață prietenoasă cu utilizatorul. Multimedia a devenit cea mai importantă cale de a realiza pagini pe Internet acum când se dezvoltă din ce în ce mai mult aplicații de e-commerce, de e-educație, e-business, e-taxe, informații, etc. Noile cerințe în materie de media nu sunt legate doar de performanța sunetului și a graficii, ci și de dezvoltarea unor proiecte de nivel înalt care să satisfacă cât mai bine cerințele din ce în ce mai pretențioase ale clientului.

### **2 Generarea imaginilor folosind PHP**

Printre alte lucruri utile, cu PHP se pot genera imagini, deoarece acest limbaj conține funcții de manipulare a imaginilor, funcții de crearea imaginilor precum și de aplicare a unor efecte.

Pentru a putea utiliza funcțiile de manipulare a imaginilor trebuie să avem acces la biblioteca grafică GD (<http://boutell.com/gd/>). Unele pachete PHP (pachetele începând de la versiunea 4.3.0) au inclusă această bibliotecă, dar este limitată la lucrul cu fișiere grafice cu format GIF, iar pentru a avea suport pentru formatele de imagini JPEG și PNG se pot descărca de pe adresele <ftp://ftp.uu.net-graphics/jpeg/> respectiv de pe adresa <http://libpng.org/pub/png/libpng.html> biblioteci suplimentare.

După cum se știe formatul JPEG este utilizat pentru o conversie a imaginii cu multe culori. Acest format imprimă și o compresie a imaginii, pentru a salva imaginea ca un fișier mai mic, astfel unele calități ale imaginii se pierd. Pentru ca ochiul uman nu poate tolera unele din aceste pierderi de calitate, formatul JPEG nu poate fi utilizat, de exemplu, pentru desenarea liniilor sau a textului. De aceea cel mai utilizat format pentru imagini în paginile Web este formatul JPEG.

În general crearea unei imagini se realizează în mai mulți pași. Mai întâi se creează o „pânză”, o suprafață de desenare, pe care se va aplica ulterior imaginea. Această suprafață de desenare este de fapt un identificator de imagine nou creat (cu ajutorul funcției `ImageCreate()` astfel `$img = ImageCreate($naltime_imag, $latime_imag)`) sau o altă imagine obținută prin modificarea unei imagini existente (cu ajutorul funcțiilor `ImageCreateFromPNG()`, `ImageCreateFromJPEG()`, `ImageCreateFromGIF()` utilizate pentru crearea unei imagini cu format PNG, JPEG respectiv GIF).

După formarea pânzei de desenare trebuie să definim culoarea de desenare pentru fiecare element al desenului ( `$culoare = ImageColorAllocate($imag, 255, 255, 255)`) după care vom crea conturul desenului sau vom imprima text (în funcție de ce vrem să facem). După ce imaginea va fi gata, trebuie să trimitem fluxul de date reprezentând imaginea către un browser folosim funcțiile `ImageGif()`, `ImageJpeg()`, `ImagePng()`, în funcție de forma sub care vrem să salvăm această imagine, trimițând mai întâi antetul corespunzător cu ajutorul funcției `header()` după care vom elibera zona de memorie alocată la crearea imaginii cu ajutorul funcției `ImageDestroy()`.

#### a. Desenarea liniilor și a formelor

Funcțiile utilizate pentru desenarea liniilor și a formelor sunt următoarele:

- `ImageLine($imag, $x_initial, $y_initial, $x_final, $y_final, $culoare)` –funcție pentru desenarea unei linii
- `ImageRectangle($imag, $x_initial, $y_initial, $x_final, $y_final, $culoare)` și `ImageFilledRectangle($imag, $x_initial, $y_initial,`

$\$x\_final$ ,  $\$y\_final$ ,  $\$culoare$ ) – funcții pentru desenarea unui dreptunghi gol respectiv a unui dreptunghi umplut

• *ImagePolygon*( $\$imag$ ,  $\$sir\_puncte$ ,  $\$nr\_varfuril$ ,  $\$culoare$ ) și *ImageFilledPolygon*( $\$imag$ ,  $\$sir\_puncte$ ,  $\$nr\_varfuril$ ,  $\$culoare$ ) – funcții pentru desenarea unui poligon gol, respectiv a unui poligon plin

• *ImageArc*( $\$imag$ ,  $\$x\_initial$ ,  $\$y\_initial$ ,  $\$inaltime$ ,  $\$latime$ ,  $\$unghi\_initial$ ,  $\$unghi\_final$ ,  $\$culoare$ ) – funcție pentru desenarea unui arc de cerc

• *ImageEllipse*()

unde:

- $\$imag$  este variabila care identifică imaginea creată anterior;
- $\$x\_initial$  și  $y\_initial$  sunt coordonatele de început ale formei ce urmează a fi desenată;
- $\$x\_final$ ,  $\$y\_final$  sunt coordonatele finale, acolo unde e cazul, a formei desenate;
- $\$culoare$  este o variabilă care păstrează culoarea definită ca în exemplul de mai sus;
- $\$font$  este un număr între 1 și 5 care desemnează fontul cu care va fi scris textul  $\$text$ ;
- $\$sir\_puncte$  este un șir care păstrează coordonatele vârfurilor poligonului;
- $\$nr\_varfuril$  este variabila care păstrează numărul de vârfuri ale poligonului;
- $\$inaltime$  păstrează înălțimea arcului de cerc;
- $\$latime$  păstrează lățimea arcului de cerc;
- $\$unghi\_initial$ ,  $\$unghi\_final$  păstrează unghiul inițial, respectiv unghiul final al arcului de cerc;

#### b. Adăugarea textului în modul grafic

Textul poate fi adăugat la o imagine atât caracter cu caracter cât și ca șir de caractere. Pentru acestea avem următoarele funcții:

- *ImageString*( $\$imag$ ,  $\$font$ ,  $\$x\_initial$ ,  $\$y\_initial$ ,  $\$text$ ,  $\$culoare$ )
- *ImageStringUp*( $\$imag$ ,  $\$font$ ,  $\$x\_initial$ ,  $\$y\_initial$ ,  $\$text$ ,  $\$culoare$ )- folosită pentru șiruri verticale

- ***ImageChar(\$imag, \$font, \$x\_initial, \$y\_initial, \$character, \$culoare)*** – această funcție, dacă este apelată într-un ciclu, poate afișa un șir de caractere, caracter cu caracter
- ***ImageCharUp(\$imag, \$font, \$x\_initial, \$y\_initial, \$character, \$culoare)*** – această funcție desenează un caracter vertical

### c. Funcții pentru manipularea imaginilor

Există mai multe funcții utilizate pentru manipularea dimensiunii imaginii, culorii, a fontului, a stilului, etc. Aceste sunt:

- ***ImageFill(\$imag, \$x\_initial, \$y\_initial, \$culoare)*** - funcție utilizată pentru umplerea unei forme
- ***GetImageSize("image.gif")*** - funcție care întoarce un șir de indecși numerici, cu primul element reprezentând lățimea imaginii în pixeli, al doilea element reprezentând înălțimea imaginii, al treilea reprezentând tipul imaginii (1 – GIF, 2- JPEG, 3 – PNG), iar al patrulea element reprezentând înălțimea și lățimea pe care o poate lua șirul introdus între tagurile HTML pentru imaginea dată
- ***ImageColorArt(\$imag, \$x\_pixel, \$y\_pixel)*** – funcție care preia culoare pixelului de coordonate *x\_pixel*, *z\_pixel* și care vor fi transmise funcției ***ImageColorsForIndex()*** care va converti aceasta culoare într-o culoare de forma RGB
- ***ImageColorsTotal(\$imag)*** – returnează numărul total de culori folosite în imaginea *imag*
- ***ImageColorTransparent()*** – aplică transparență background-ului
- ***ImageCopy(\$img\_sursa, \$img\_dest, \$x\_dest, \$y\_dest, \$x\_sursa, \$y\_sursa, \$latime\_sursa, \$inaltime\_sursa)*** – copiază imaginea sursă peste imaginea destinație specificând noua dimensiune a imaginii
- ***ImageLoadFont("font1.ft")***, ***ImagePsLoadFont("font2.ft")***, ***ImagePsFreeFont(index\_font)***, ***ImagePsLoadFont("font3.ps")*** - încarcă diferite fonturile

Iată un mic program PHP care folosește unele din aceste funcții de manipulare a imaginilor:

```
<?php  
header("Content-type: image/gif");  
$sir = "Sir scris pe verticala caracter cu caracter";
```

```
$sir2="Alt mod de scriere";

$imagine = ImageCreate(600, 700);

$bgcol = ImageColorAllocate($imagine, 200,200,255);
$albastru_d = ImageColorAllocate($imagine, 150,150,255);
$rosu_d = ImageColorAllocate($imagine, 255, 150, 150);
$verde_d= ImageColorAllocate($imagine, 150, 255,150);
$alb = ImageColorAllocate($imagine, 255, 255, 255);
$negru= ImageColorAllocate($imagine, 0, 0 ,0);
$rosu = ImageColorAllocate($imagine, 250, 0 ,0);
$verde = ImageColorAllocate($imagine, 0, 250 ,0);
$albastru = ImageColorAllocate($imagine, 0, 0 ,250);

//scris pe verticală
for($i=0,$y=strlen($sir)*15+20;$i<strlen($sir); $i++, $y-=15)
{
ImageCharUp($imagine, 5, 10, $y, $sir[$i], $negru);
}

//scris pe orizonta[
ImageString($imagine, 5, 200, 30, "Modurl grafic in PHP",
$negru);

ImageRectangle($imagine, 180, 15, 410,50, $negru);
$varfuri_p=array(150,250,275,150,400,250,400,350,275,450,150,
350);

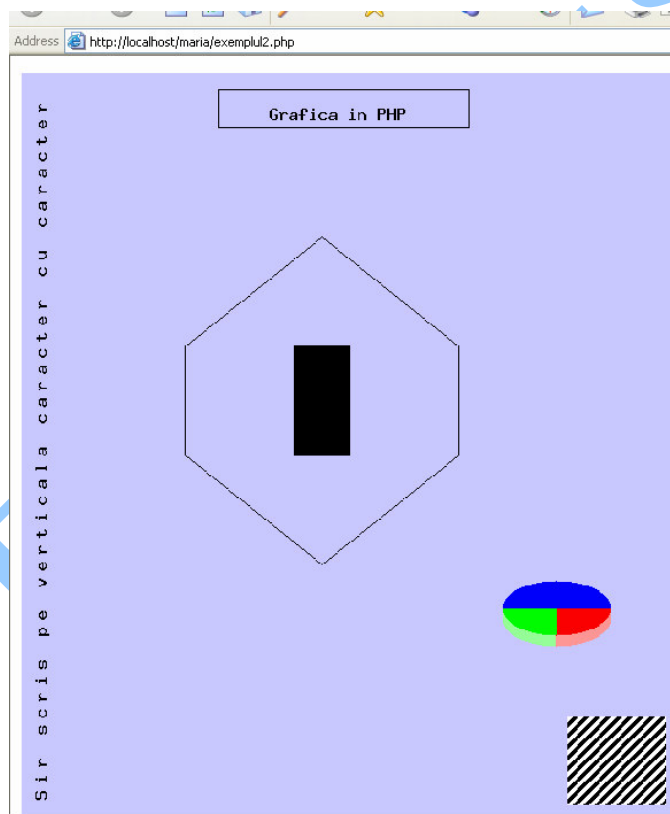
ImageColorTransparent($imagine, $alb);
ImageFilledRectangle($imagine, 250, 250, 300, 350, $negru);
ImagePolygon($imagine, $varfuri_p, 6, $negru);

//desenare zona umbra
for($i=500; $i>490; $i--){
ImageFilledArc($imagine, 490, $i, 100, 50, 0, 90,
$rosu_d, IMG_ARC_PIE);
ImageFilledArc($imagine, 490, $i, 100, 50, 91, 180,
$verde_d, IMG_ARC_PIE);
ImageFilledArc($imagine, 490, $i, 100, 50, 181, 360,
$albastru_d, IMG_ARC_PIE);
}

//desenare grafic tip diagrama
ImageFilledArc($imagine, 490, 490, 100, 50, 0, 90, $rosu,
IMG_ARC_PIE);
ImageFilledArc($imagine, 490, 490, 100, 50, 91, 180, $verde,
IMG_ARC_PIE);
```

```
ImageFilledArc($image, 490, 490, 100, 50, 181, 360,  
$albastru, IMG_ARC_PIE);  
  
$stil = array($alb, $alb, $alb, $alb, $alb,  
$negru, $negru, $negru, $negru, $negru);  
ImageSetStyle($image, $stil);  
ImageFilledRectangle($image, 500, 590, 590, 670,  
IMG_COLOR_STYLED);  
  
ImageGif($image);  
ImageDestroy($image);  
?>
```

iar efectul acestui cod este ilustrat in figura următoare:



Deoarece limbajul PHP are o compatibilitate ridicată cu foarte multe alte limbaje/programe cum ar fi Java, JavaScript, Flash, etc, și deoarece

unele din aceste limbaje sunt mai eficiente în prelucrarea imaginilor se pot crea și manipula imagini cu aceste limbaje după care pot fi intercalate într-o aplicație PHP. În PHP crearea imaginilor se face pas cu pas prin cod, nu se poate folosi mouse-ul pentru a desena, nu există instrumente de umplere a spațiilor predefinite, etc, în timp ce în alte programe de desenare există și, deci, desenarea sau manipularea imaginilor se face mai ușor.

### **Bibliografie**

- [GB00] **Jay Greenspan, Brad Bulger** - *MySql/Php Databases application*, M&T Books, 2000
- [Mel05] **Julie C. Meloni** - *PHP, MySQL și Apache*, Ed. Corint, București, 2005
- [Joh96] **David Sklar, Adam Trachtenberg** - *PHP Cookbook*, O'Reilly, 2002
- [Hug01] **Sterling Hughes** - *PHP Developer's Cookbook, Second Edition*, Sams Publishing, 2001

Tibiscus