**Redimensionar imágenes**

<http://www.imaginanet.com/blog/crear-thumbnails-miniaturas-con-php-al-vuelo.html>

# **Crear thumbnails (miniaturas) con php al vuelo.**

*Publicado el 11 de Julio de 2011 en las categorías*[*PHP*](http://www.imaginanet.com/blog/crear-thumbnails-miniaturas-con-php-al-vuelo.html?cat=php)*,*[*XHTML*](http://www.imaginanet.com/blog/crear-thumbnails-miniaturas-con-php-al-vuelo.html?cat=xhtml)

Para reducir el ancho de banda en nuestro servidor, a costa de cargar la cpu un poco más, podemos redimensionar las imágenes que estamos sirviendo con un sencillo script de PHP. Vamos a ver dos métodos, uno con la librería gd y otro llamando a imagemagick.

## Librería GD:

Usaremos un archivo php para hacer la carga de las imágenes, lo llamaremos thumb.php y lo colocaremos en la raíz de nuestra web. Lo llamaremos de la siguiente forma:

[?](http://www.imaginanet.com/blog/crear-thumbnails-miniaturas-con-php-al-vuelo.html)

|  |  |
| --- | --- |
| 1 | <img src="/thumb.php?file=/public/image1.png&width=100" alt="miniatura" /> |

Le pasamos 2 parámetros al script:

**file**: la ruta a la imagen que queremos redimensionar.

**width**: la nueva anchura.

Lo primero que tenemos que hacer el calcular las nuevas dimensiones de la imagen, en nuestro ejemplo estamos marcando la anchura, y la altura se calcula a partir de ésta. Cuando ya tenemos las nuevas dimensiones vemos en qué formato está y luego escribimos las cabeceras correspondientes y la mostramos. El código del archivo thumb.php sería el siguiente:

[?](http://www.imaginanet.com/blog/crear-thumbnails-miniaturas-con-php-al-vuelo.html)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36 | $file = $\_GET["file"];  $width = $\_GET["width"];    // Ponemos el . antes del nombre del archivo porque  estamos considerando que la ruta está a partir del archivo thumb.php  $file\_info = getimagesize("." . $file);  // Obtenemos la relación de aspecto  $ratio = $file\_info[0] / $file\_info[1];    // Calculamos las nuevas dimensiones  $newwidth = $width;  $newheight = round($newwidth / $ratio);    // Sacamos la extensión del archivo  $ext = explode(".", $file);  $ext = strtolower($ext[count($ext) - 1]);  if ($ext == "jpeg") $ext = "jpg";    // Dependiendo de la extensión llamamos a distintas funciones  switch ($ext) {          case "jpg":                  $img = imagecreatefromjpeg("." . $file);          break;          case "png":                  $img = imagecreatefrompng("." . $file);          break;          case "gif":                  $img = imagecreatefromgif("." . $file);          break;  }  // Creamos la miniatura  $thumb = imagecreatetruecolor($newwidth, $newheight);  // La redimensionamos  imagecopyresampled($thumb, $img, 0, 0, 0, 0,  $newwidth, $newheight, $file\_info[0], $file\_info[1]);  // La mostramos como jpg  header("Content-type: image/jpeg");  imagejpeg($thumb, null, 80); |

## Imagemagick

Si usamos imagemagick tenemos las ventajas de ahorrar consumo de memoria y los inconvenientes de que tenemos que tenerlo instalado en nuestro servidor. En caso de tenerlo disponible, el script resultante sería muy parecido al anterior

[?](http://www.imaginanet.com/blog/crear-thumbnails-miniaturas-con-php-al-vuelo.html)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | $file = $\_GET["file"];  $width = $\_GET["width"];    // Ponemos el . antes del nombre del archivo porque  estamos considerando que la ruta está a partir del archivo thumb.php  $file\_info = getimagesize("." . $file);  // Obtenemos la relación de aspecto  $ratio = $file\_info[0] / $file\_info[1];    // Calculamos las nuevas dimensiones  $newwidth = $width;  $newheight = round($newwidth / $ratio);    // No necesitamos sacar la extensión ya que el comando convert  ya detecta el formato  header("Content-type: image/jpeg");  passthru("convert -scale ".($newwidth)."x".  ($newheight)."! "./".$file."" JPG:-"); |

<http://code.runnable.com/UnF-tFdudNt1AABt/how-to-resize-an-image-using-gd-library-for-php>

<?php

/\*

\* PHP GD

\* resize an image using GD library

\*/

// File and new size

//the original image has 800x600

$filename = 'images/picture.jpg';

//the resize will be a percent of the original size

$percent = 0.5;

// Content type

header('Content-Type: image/jpeg');

// Get new sizes

list($width, $height) = getimagesize($filename);

$newwidth = $width \* $percent;

$newheight = $height \* $percent;

// Load

$thumb = imagecreatetruecolor($newwidth, $newheight);

$source = imagecreatefromjpeg($filename);

// Resize

imagecopyresized($thumb, $source, 0, 0, 0, 0, $newwidth, $newheight, $width, $height);

// Output and free memory

//the resized image will be 400x300

imagejpeg($thumb);

imagedestroy($thumb);

?>

POO:

<http://code.tutsplus.com/tutorials/image-resizing-made-easy-with-php--net-10362>

Extensión PHPThumb:

<http://phpthumb.sourceforge.net/>