

## Códigos QR en PHP



Creo que hoy día todos sabemos lo que son los códigos QR (Quick Response). Es una manera de codificar datos en una imagen bidimensional, cómo la que aparece a la izquierda, de forma que pueden ser leídos con un dispositivo adecuado, obteniendo el texto original. Son la versión bidimensional del tradicional código de barras. Se pueden emplear para tarjetas identificativas, marcaje de productos comerciales, entradas de espectáculos, controles de acceso, etc.

Si quieres aprender los conceptos teóricos de los código QR te sugiero que visites [esta página](#), donde encontrarás abundante información.

Leer un código QR es muy fácil. Existen dispositivos especializados que leen este tipo de códigos y salen a un precio muy económico, lo que los hace aptos para tiendas. Sin embargo, más fácil aún, es instalar en un smartphone la aplicación adecuada, que los lee y descodifica, obteniendo una URL o el texto que sea que se haya codificado.

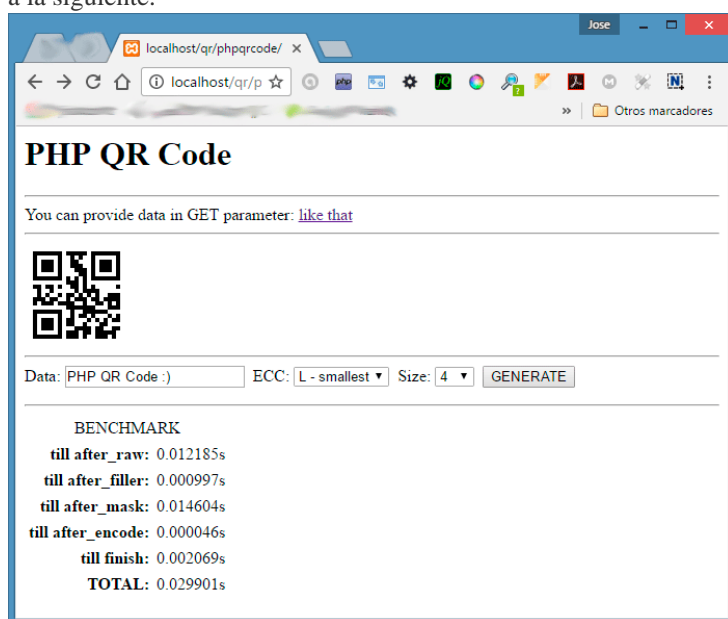
### CREAR CÓDIGOS QR CON PHP

En este artículo vamos a aprender a usar una librería especializada de PHP para generar códigos QR de forma flexible, rápida, y fácilmente adaptable a las necesidades de nuestro proyectos. Para ello, crearemos un proyecto en nuestro localhost, donde haremos nuestras pruebas. Vamos a recurrir a la librería PHPQRCode, que puedes descargar en [este enlace](#). Almacena el comprimido dentro de la carpeta de tu proyecto en tu ordenador, y descomprímelo ahí.

Dentro de localhost tienes el directorio de tu proyecto (que, en mi caso, he llamado `qr`) y, dentro de este, a su vez, tienes una carpeta llamada `phpqrcode`, que has extraído del archivo comprimido que te descargaste. En este directorio está el fichero principal de la librería, llamado `qrlib.php`. Además, hay varios scripts PHP con distintas funcionalidades del código. También hay un archivo `index.php` con un formulario para llamar a la librería y facilitarte que hagas tus pruebas iniciales.

### LA PRIMERA PRUEBA

Bien. Ya que los autores nos ofrecen un archivo de prueba, vamos a usarlo. Una vez levantado tu servidor en tu máquina local, entra en el navegador, teclea <http://localhost/qr/phpqrcode> en la barra de direcciones, y pulsa la tecla <ENTER>. Obtendrás una página similar a la siguiente:



Vemos un código QR de muestra en la parte superior izquierda. Si tienes en tu smartphone una app lectora, úsala para leerlo. Tu teléfono mostrará que el contenido es [PHP QR Code](#) :).

Ahora observa el formulario que hay debajo de la imagen. En el primer campo, llamado [Data](#) escribe lo que quieras. Puedes poner, por ejemplo, la URL de alguna pagina que desees. El campo llamado [ECC](#) se refiere a la calidad con la que quieres que se genere tu QR. Cuanta más calidad pongas, más facil le será a la app del móvil (o dispositivo lector que emplees) leer el QR, pero también tendrá un mayor peso la imagen con el QR, y requerirá mayor tiempo de proceso crearlo. Las calidades altas solo se emplean si tu código debe ser leído en condiciones adversas (al aire libre, con el sol reflejando, por ejemplo). En el campo [Size](#) elige el tamaño que desees. Pulsa el botón [GENERATE](#) y verás que se genera un código QR. Si lo lees con tu móvil verás que obtienes el contenido que pusiste en [Data](#).

### GENERAR TUS PROPIOS QR

Para generar tus propios códigos QR debes incluir, en tus scripts la librería [qrlib.php](#) que, asu vez, incluirá los scripts necesarios. En la ruta base de mi directorio incluyo un [index.php](#) con la siguiente instrucción:

```
include("phpqrcode/qrlib.php");
```

Evidentemente, tengo que tener en cuenta la ruta donde está la librería que quiero incluir con respecto a mi script, por supuesto.

A continuación incluyo una llamada al método [QRcode::png\(\)](#) pasándole los siguientes argumentos:

El contenido a codificar (una URL, una frase, un identificador de una persona, lo que sea que quiera codificar en el QR).

El fichero (y en su caso la ruta) en la que grabaré la imagen del QR. Este método graba el QR como una imagen en disco.

Un indicativo de la calidad con la que deseo que se cree la imagen. Puede ser:

L. La más baja.

M. Calidad media.

Q. Buena calidad.

H. La más alta calidad posible.

Luego incluyo una referencia al tamaño del QR. Esta puede ser un valor entre 1 y 10. El QR será un cuadrado cuyo lado medirá  $33 * n$  píxeles, donde  $n$  es el valor que incluyo aquí. Así, si pongo un valor 5, la imagen del QR tendrá 165 píxeles de lado ( $33*5$ ).

El margen blanco que queda alrededor del propio QR en la imagen. Es un factor de 5, es decir, si ponemos el valor 2, quedará un margen en blanco de 10 píxeles por cada lado.

La instrucción será algo así:

```
QRcode::png("http://eldesvadejose.com", "eldesvadejose.png", "Q", 4, 2);
```

Y ya tengo la imagen del QR creada, y grabada en el disco (por supuesto, si tengo permisos de escritura sobre el directorio de destino). Ya solo tengo que volcarla a pantalla, así;

```
echo '';
```

Cómo ves, fácil y rápido. Y, aunque lo expuesto en este artículo cubre el 90 % de las necesidades que puedas tener respecto a códigos QR, si quieres más detalles deberías visitar la página oficial del proyecto, ya que, de vez en cuando, sacan actualizaciones interesantes, y proporciona documentación muy detallada. La encontrarás en [este enlace](#).