

Cómo hacer una preview de una imagen antes de subirla al servidor



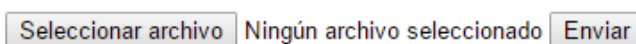
A la hora de desarrollar un **portal web** es muy habitual encontrarnos con formularios que nos permiten subir imágenes al servidor. Un claro ejemplo lo tenemos en redes sociales como **Facebook** o **Twitter**, donde podemos personalizar nuestro perfil con nuestras imágenes. Lo que pretendemos con este White Paper es mostraros un ejemplo de código que permite hacer una preview de la imagen antes de ser enviada al servidor.

¿Por qué hacer una preview?

Cuando hablamos de preview nos estamos refiriendo a la posibilidad de que el usuario vea la imagen que quiere enviar al **servidor** antes de hacerlo, en vez de mostrar únicamente el nombre del archivo a subir, que es la forma por defecto utilizada. Con esta acción permitimos que los usuarios estén seguros de que la imagen es la correcta y de no ser así, poder cambiarla sin tener que repetir el proceso de nuevo. Puede que esto suponga un poco más de trabajo a la hora de crear el portal, pero estaremos añadiendo un valor extra a nuestro sitio.

Para conseguir esta funcionalidad, a lo largo de nuestro ejemplo utilizaremos JavaScript, **jQuery**, PHP y HTML. Empecemos con el desarrollo del código.

Archivo index.html



Lo primero de todo será crear el campo de subida de la imagen. Para ello, utilizaremos un formulario creado en HTML que únicamente tendrá un campo input de tipo **"file"** y un botón que ejecutará el archivo PHP encargado de subir la imagen. El código encargado de pintar este pequeño formulario de ejemplo es el siguiente:

```
<form method="post" action="upload.php" enctype="multipart/form-data"
id="uploadForm">
  <input type="file" name="file" id="file" />
  <input type="submit" name="Enviar" value="Enviar"/>
</form>
```

Como podéis ver se trata de un código muy sencillo con los dos campos comentados anteriormente que están dentro de un formulario que ejecuta el script **"upload.php"**, encargado de realizar la subida de la imagen.

Archivo upload.php

En el archivo **upload.php**, lo que tendremos será el código PHP encargado de subir la imagen al directorio correspondiente:

```
<?php
if(isset($_POST['Enviar']) && !empty($_FILES['file']['name'])) {

if(move_uploaded_file($_FILES['file']['tmp_name'], "uploads/".$_FILES['file']['name'])) {
echo Archivo subido correctamente.';
} else {
    echo Ocurrió algunos problemas. Inténtelo más tarde.';
}
}
```

Aunque se trata de un código muy sencillo para aquellos que estén acostumbrados a trabajar con PHP, vamos a explicar los puntos más importantes de él. Lo primero que hacemos es detectar si por POST se ha enviado el valor del campo cuyo nombre es **"Enviar"** y que hace referencia al botón del formulario. Con esto nos estamos asegurando de que se haya pulsado el botón. También miramos que se haya elegido una imagen para subir. Todo esto lo hacemos con la primera línea de código que os dejamos a continuación.

```
if(isset($_POST['Enviar']) && !empty($_FILES['file']['name']))
```

Si se cumplen las dos premisas anteriores, entonces se ejecutará el método **"move_uploaded_file"** de PHP y que se encarga de mover la imagen a la carpeta que indiquemos dentro del **alojamiento web**. A este método se le pasa dos parámetros. Como primer parámetro el nombre temporal de la imagen a subir y como segundo la ruta donde queremos que se guarde. En nuestro ejemplo, le decimos que lo haga en una carpeta de nombre **"uploads"**. El código que hace todo esto es el siguiente.

```
if(move_uploaded_file($_FILES['file']['tmp_name'], "uploads/".$_FILES['file']['name'])) {
echo Archivo subido correctamente.';
} else {
    echo Ocurrió algunos problemas. Inténtelo más tarde.';
}
```

Si la imagen se ha subido correctamente, se mostrará el mensaje de "Archivo subido correctamente". Si por el contrario no se puede subir, entonces se mostraría el mensaje: "Ocurrió algunos problemas. Inténtelo más tarde".

Código JavaScript

Ahora es el turno del código encargado de realizar el preview de la imagen que queremos subir. Para ello, utilizaremos algunas funciones jQuery que nos facilitará el trabajo. Como hemos hecho hasta ahora, a continuación os dejamos el código encargado de realizar esto.

```
function filePreview(input) {
  if (input.files && input.files[0]) {
    var reader = new FileReader();
    reader.readAsDataURL(input.files[0]);
    reader.onload = function (e) {
      $('#uploadForm + img').remove();
      $('#uploadForm').after('');
    }
  }
}
```

El código anterior se trata de una función que se encarga de hacer el preview de la imagen a subir. Para ello, lo primero que hacemos es comprobar que existe una imagen a subir. Esta comprobación se realiza con la siguiente instrucción.

```
if (input.files && input.files[0])
```

Si existe la imagen, entonces lo que hacemos será crear un objeto de tipo "FileReader" y al que hemos llamado "reader" aunque se le puede poner cualquier nombre.

```
var reader = new FileReader();
```

Con el método "readAsDataURL" leeremos el contenido del archivo pasado por parámetro, en este caso, la imagen a subir.

```
reader.readAsDataURL(input.files[0]);
```

El siguiente paso será detectar cuando se ha cargado por completo la imagen. Para ello, tiraremos de la función "onload" de jQuery.

```
reader.onload = function (e) {
```

Una vez cargada, lo primero que haremos será eliminar cualquier etiqueta "img" que pudiera existir dentro del formulario para que únicamente se muestre la imagen que queremos subir. Si no hacemos esto, en el caso de habernos equivocado y querer subir otra imagen diferente, se mostraría tanto la imagen errónea como la correcta, una junto a la otra. El código encargado de hacer esto es el siguiente:

```
$('#uploadForm + img').remove();
```

Por último, solo falta hacer que se muestre la imagen que vamos a subir, cosa que conseguimos con la siguiente línea de código.

```
$('#uploadForm').after('');
```

Para llamar a esta función encargada de hacer el preview, hay que detectar que se ha seleccionado una imagen en el campo "File" de nuestro formulario. El código encargado de hacer esto es el siguiente.

```
$('#file').change(function () {  
    filePreview(this);  
});
```

Con esto ya tendríamos listo nuestro código para mostrar la preview de la imagen que vamos a subir al servidor.