

Prevalidar ficheros en el lado del cliente



A menudo nos encontramos con que, en un formulario, debemos incluir un campo que permita al usuario enviar archivos al servidor (imágenes, documentos pdf, etc). En los formularios tradicionales, una vez enviado el formulario se comprobaban los ficheros en el servidor (tipo de archivo, peso, etc). Esto supone un problema por cuanto que los archivos deben ser, necesariamente, enviados al servidor (con el consumo de recursos que esto conlleva) y, una vez allí, si no son adecuados, hay que volver a cargar el formulario, recuperando los datos ya enviados para que el usuario no deba teclearlos de nuevo.

El problema se ve agravado con los campos de tipo `file` de HTML 5. Hasta HTML 4, estos campos sólo podían recibir un archivo cada uno, con lo que podíamos limitar el número de archivos que le permitíamos enviar al usuario, simplemente poniendo un campo de tipo `file` por cada archivo que esperamos. Esto era bastante engorroso. Con HTML 5 los campos de tipo `file` pueden llevar el atributo `multiple`, lo que permite, en un solo campo, enviar varios archivos. No obstante, debemos prever un mecanismo para limitar el número máximo de archivos.

La solución es hacer una prevalidación en el lado del cliente ANTES de que el formulario se envíe al servidor. Gracias a JavaScript y jQuery esa tarea es muy fácil de llevar a cabo con rapidez, comodidad y eficiencia.

Por supuesto, obvio es decirlo, en nuestra página deberemos incorporar jQuery. Supongamos que tenemos un formulario con un campo file, configurado con el atributo `multiple`, para que el usuario pueda seleccionar varios archivos a la vez, así:

```
<form id='formulario' action='recibir_ficheros.php' method='post' enctype='multipart/form-data'>
<input type='file' id='archivosSeleccionados' name='archivosSeleccionados[]' multiple>
<br>
<input type='submit' id='botonEnviar' value='Enviar ficheros'>
</form>
```

Esto es un formulario de lo más simple, sin nada de particular. Ahora veamos el código jQuery. Lo primero que hacemos es detectar la pulsación del botón de envío, y cancelarla para hacer la prevalidación, así:

```
$('#botonEnviar').on('click', function(e){
e.preventDefault(); // De momento anulamos el envío del formulario
var erroresEnArchivos = false; // Para comprobar si hay errores
var maximoDeArchivos = 4; // El número máximo de archivos que se podrán enviar
var pesoMaximoPorArchivo = 100000; // El peso máximo en bytes por archivo.
var matrizDeTiposAdmitidos = new Array(
'image/jpeg',
'image/png'
); // Los tipos de archivos que le vamos a admitir al usuario.
/* A partir de aquí, se identifican los archivos y se hace la prevalidación. */
...
});
```

Ya lo tenemos preparado, con los valores que hemos elegido para este ejemplo como máximo número de archivos a enviar, tipos de archivos admitidos y el peso máximo por archivo, en bytes.

A continuación vamos a hacer la comprobación de que los archivos que haya seleccionado el usuario se ajusten a estos parámetros. Si alguno de los valores excede el máximo permitido, pondremos en `true` el flag de detección de errores, e impediremos, definitivamente, el envío. Además, podremos elegir si le mostramos un aviso al usuario, o tomamos alguna medida específica.

```
/* Lo primero es crear una matriz con todos los archivos que el usuario haya incluido en el campo file */
```

```
var archivosSeleccionados = $('#archivosSeleccionados')[0]['files'];
// Si se excede el número de archivos, marcamos que hay error.
if (archivosSeleccionados.length > maximoDeArchivos) erroresEnArchivos = true;
if (!erroresEnArchivos){ // Si no hay error, seguimos comprobando
for (var archivo in archivosSeleccionados){
  if (archivo != parseInt(archivo)) continue;
  if (archivosSeleccionados[archivo]['size'] > pesoMaximoPorArchivo){
    erroresEnArchivos = true;
    break;
  }
  if (matrizDeTiposAdmitidos.indexOf(archivosSeleccionados[archivo]['type']) < 0){
    erroresEnArchivos = true;
    break;
  }
}
}
```

/* Si hay un error, se avisa de ello. Si no, se envía el formulario. */

```
if (erroresEnArchivos){
  alert ("Se han producido errores");
} else {
  $('#formulario').submit();
}
```

Lo primero es almacenar en una matriz de javascript todos los archivos que el usuario haya seleccionado en el campo [file](#). En jQuery esto es muy sencillo, así:

```
var archivosSeleccionados = $('#archivosSeleccionados')[0]
```

A pesar de esta simplicidad, algunas personas desconocen este recurso, por lo que lo he resaltado en el fragmento de código anterior. A continuación determinamos el número de archivos que tiene esta matriz, y comprobamos que no se haya excedido el número máximo de archivos que le autorizamos al usuario a enviar y que, en este ejemplo, hemos fijado en 4.

```
if (archivosSeleccionados.length > maximoDeArchivos) erroresEnArchivos = true;
```

Si, hasta ahora, no se ha producido ningún error, vamos a comprobar cada archivo. Para ello emplearemos una variable de bucle que actuará como índice de la matriz de archivos, permitiéndonos acceder a cada elemento. A esta variable, en el ejemplo, la hemos llamado [archivo](#):

```
for (var archivo in archivosSeleccionados){
```

Lo primero que hacemos es comprobar que el índice [archivo](#) sea numérico. Esto se hace porque, aunque la matriz tiene un elemento por cada archivo seleccionado, internamente aparecen elementos, sin un índice numérico, que tienen que ver con la forma en que el navegador contempla la matriz de archivos. En la práctica, esto significa que podríamos tener elementos "fantasma" que no se cuentan como tales elementos y no tienen un índice numérico, pero, a la hora de recorrer la matriz elemento a elemento aparecen, induciendo a confusión. Recuerda que SÓLO los archivos que el usuario ha incluido en el campo [file](#) tienen un índice numérico. Por lo tanto, si durante el recorrido de la matriz encontramos un "supuesto" contenido que no tiene tal índice, debemos ignorarlo:

```
if (archivo != parseInt(archivo)) continue;
```

El siguiente paso es usar la propiedad [size](#) de cada uno de los archivos, para determinar si excede el peso máximo que hemos admitido (en el ejemplo, [100000 bytes](#)).

```
if (archivosSeleccionados[archivo]
  erroresEnArchivos = true;
  break;
}
```

Ahora nos toca determinar que el tipo del archivo (propiedad [type](#)) esté en la matriz de tipos admitidos, así:

```
if (matrizDeTiposAdmitidos.indexOf(archivosSeleccionados[archivo]
  erroresEnArchivos = true;
```

```
break;  
}
```

Y ya está. Si, al final de este proceso, no hay errores registrados, se puede enviar el formulario. El conjunto total del código, para que lo veas globalmente, ahora que ya sabemos lo que hay que hacer, es el siguiente:

```
<!DOCTYPE html>  
<html lang='es'>  
<head>  
<meta charset='UTF-8'>  
<title>Ficheros</title>  
</head>  
<body>  
<form id='formulario' action='recibir_ficheros.php' method='post' enctype='multipart/form-data'>  
<input type='file' id='archivosSeleccionados' name='archivosSeleccionados[]' multiple>  
<br>  
<input type='submit' id='botonEnviar' value='Enviar ficheros'>  
</form>  
<p> </p>  
<div id='resultados' style='overflow:visible;'></div>  
<script src='https://code.jquery.com/jquery-3.2.1.min.js'></script>  
<script language='javascript'>  
$( '#botonEnviar' ).on( 'click', function( e ) {  
    e.preventDefault(); // De momento anulamos el envío del formulario  
    var erroresEnArchivos = false; // Para comprobar si hay errores  
    var maximoDeArchivos = 4; // El número máximo de archivos que se podrán enviar  
    var pesoMaximoPorArchivo = 100000; // El peso máximo en bytes por archivo.  
    var matrizDeTiposAdmitidos = new Array(  
        'image/jpeg',  
        'image/png'  
    ); // Los tipos de archivos que le vamos a admitir al usuario.  
    /* A partir de aquí, se identifican los archivos y se hace la prevalidación. */  
    /* Lo primero es crear una matriz con todos los archivos que el usuario haya incluido en el campo file */  
    var archivosSeleccionados = $( '#archivosSeleccionados' )[ 0 ][ 'files' ];  
    // Si se excede el número de archivos, marcamos que hay error.  
    if ( archivosSeleccionados.length > maximoDeArchivos ) erroresEnArchivos = true;  
    if ( !erroresEnArchivos ) { // Si no hay error, seguimos comprobando  
        for ( var archivo in archivosSeleccionados ) {  
            if ( archivo != parseInt( archivo ) ) continue;  
            if ( archivosSeleccionados[ archivo ][ 'size' ] > pesoMaximoPorArchivo ) {  
                erroresEnArchivos = true;  
                break;  
            }  
            if ( matrizDeTiposAdmitidos.indexOf( archivosSeleccionados[ archivo ][ 'type' ] ) < 0 ) {  
                erroresEnArchivos = true;  
                break;  
            }  
        }  
    }  
    /* Si hay un error, se avisa de ello. Si no, se envía el formulario. */  
    if ( erroresEnArchivos ) {
```

```
    alert ("Se han producido errores");  
  } else {  
    $('#formulario').submit();  
  }  
});  
</script>  
</body>  
</html>
```