Jun 25 2010

Uploading Files via JQuery, Form Plugin & ASP.NET MVC

Finding a good way to do asynchronous uploads to a website has been tough. Honestly, I don't know why they haven't built the ability to upload via XHR into modern browsers yet. At any rate, I looked at a lot of solutions, most of which are built on Flash *ugh*, and I just couldn't find anything that was really solid. I finally found a clever little JQuery form plugin creatively called….. drum roll please… JQuery Form Plugin! [Website] So today I'm going to explain the proper way to use it to upload files asynchronously because there's a few special things you need to know about.

First you need a form in your View. This can be a standard submit form but you need to make sure you set the *enctype* properly. Here's what mine looks like.

```
<form id="ajaxUploadForm" action="<%= Url.Action("UploadImage", "Upload")%>" metho
  <fieldset>;
    <legend>Upload</legend>
    <input id="file" name="file" type="file" value="" />
    <input id="ajaxUploadButton" type="submit" value="Submit" />
  </fieldset>
</form>
```

Easy enough. Now make sure you've included JQuery and the JQuery Forms JavaScript into your page. Then you'll add a JavaScript function to catch the submit and deal with it properly. Here's mine:

```
$("#ajaxUploadForm").ajaxForm({
iframe: "true",
success: function (response) {
var msg = $.parseJSON($(response).val());
if (msg.status == "valid") {
fadeInPageMessage("success", msg.message);
} else if (msg.status == "invalid") {
fadeInPageMessage("error", msg.message);
}
}
});
```

So there's a few important things to note here. I set the iframe parameter to "true". This is done because you can't upload a file via an AJAX call. So what the Form Plugin does is create an iframe dynamically, then submits the form as a normal POST via that iframe.

Now, since we've done a POST through an iframe and not via an XHR (XmlHttpRequest), the return data is also going to come back through that iframe, and it's going to come back like a normal page would. This causes a lot of problems for people because normally your Controller Action would return a JsonResult which comes down to the browser as the type "application/json". Since this is coming down to that iframe the browser thinks you're downloading a file and prompts the user to pick a program to open it. This is obviously not what we want.

So how do we get our JSON result from this iframe and get it back to our javascript function so we can deal with it? Well, we have to return the data as HTML instead of JSON so the browser will not prompt the user. And since we're sending down HTML we'll follow the guidelines set by the creators of the Form Plugin and place our JSON string inside a TextArea. Here's how that's done.

We'll create a new class that extends the JsonResult class. I found this code by John Rudolf Lewis and it works fantastic:

```
public class FileUploadJsonResult : JsonResult
{
public override void ExecuteResult(ControllerContext context)
{
this.ContentType = "text/html";
context.HttpContext.Response.Write("&lt;textarea&gt;");
base.ExecuteResult(context);
context.HttpContext.Response.Write("&lt;/textarea&gt;");
}
```

Tags

AJAX ASP.NET MVC bing blog browsers C# Codeigniter coding community server Computers concepts CSS Decisions delegates development Drupal error fonts Form Plugin forms Framework generalist google hosting IIS internet iPad iPhone javascript jobs jQuery lambda layoffs linux logging Microsoft MVC mssql MS SQL Objective C php PredicateBuilder productivity SharePoint web web development

My Tweets

Amazing pic. http://t.co/xmhl05rExE

RT @GoogleFacts: California (population 37,000,000) has only 58 more bars than Wisconsin (population 5,600,000)

RT @ConanOBrien: YouTube may start charging? I guess cats are sick of working for free.

Archives

- January 2013
- May 2012
- April 2012
- March 2012
- January 2012
- July 2011
- June 2011
- March 2011
- November 2010
- October 2010
- August 2010
- June 2010
- April 2010
- March 2010
- February 2010
- January 2010
- December 2009

Web2PDF
converted by Web2PDFConvert.com

```
}
```

You use this class exactly like you would use your regular JsonResult class. As a result, here's my Controller:

```
public ActionResult UploadImage()
{
HttpPostedFileBase file = Request.Files["file"];
if (file.ContentLength &gt; 0)
{
file.SaveAs([put your path here]);
return new FileUploadJsonResult {
Data = new {
status = "valid",
message = "Upload Successful"
}
};
}
else
{
return new FileUploadJsonResult {
Data = new {
status = "invalid",
message = "No Image Selected for Upload"
}
};
}
}
```

So this code checks to see if there's a file and if so, saves it to disk and sends back a JSON response. If no file exists it sends back a separate response.

There's one final thing that's important to note. Now that we have the data in our iframe, we need to turn it back into a JSON object. If you go back at our client side JavaScript you'll see the following code:

```
success: function (response) {
var msg = $.parseJSON($(response).val());
...
```

This takes the response, which is just a TextArea with some content, and using the val() function gets our JSON string. Then it passes it on to the parseJSON function which turns that string back into an object and assigns it to the msg variable. Now we can manipulate msg any way we see fit.

So in closing, the way the Form Plugin works is that it submits your form through an iframe. You process that submission, formulate your JSON string, wrap it in in a TextArea, and send it back to the browser as HTML. Then you strip the JSON string out of that HTML, turn it back into an object and do with it what you please.

I hope this clarifies the process for some of you and helps you to write great, Flash free upload pages.

Posted by Dana Pellerin at 9:41 am Tagged with: ASP.NET MVC, Form Plugin, jQuery, Uploader

6 Responses to "Uploading Files via JQuery, Form Plugin & ASP.NET MVC"

1. **Wayne Brantley** says:
   July 26, 2010 at 5:50 am

   modern browsers DO support XHR for file uploading.
   Not sure how to received this in MVC, but at least the support is finally there!

   http://valums.com/ajax-upload/

   > **Dana** says:
   > July 26, 2010 at 5:51 am
   >
   > 
   >
   > Sweet, thanks for the link Wayne!

2. **Roman** says:
   October 6, 2010 at 12:15 pm

   Great thanks!

3. **shyam** says:
   February 4, 2011 at 6:04 am

August 2009

Meta

- Log in
- Entries RSS
- Comments RSS

Web2PDF
converted by Web2PDFConvert.com

hi,

When i tried this sample, page gets navigated. I need to render the json result in the same page without navigating.

Pls help me out!!!!!!!
Thanks in advance

4. **Carlo** says:
March 1, 2011 at 6:57 am

GREAT!!!! You saved me a lot of time! Thanks!

5. **Adrian** says:
April 16, 2011 at 12:37 pm

Awesome post! Thanks for taking the time to write that up, it really helped me out.

Sorry, the comment form is closed at this time.

WebFonts Rock! Linode Rocks!

© 2012 64 Square Feet

Suffusion theme by Sayontan Sinha

Web2PDF
converted by Web2PDFConvert.com