Uploading multiple files from a form with Zend Framework (ZF) / Storing uploaded zf files with unique name

Author : admin

Here is the code basis I used to implement a form file uploader in Zend Framework (ZF).
I tried to write the code as self explanatory as possible, hope I achieved my goal. If you have to solve the task I believemy code can help you out. Check out the code:

```
  // open Zend Http file transfer stream
$file_upload = new Zend_File_Transfer_Adapter_Http();
// Returns all known internal file information
$files = $file_upload->getFileInfo();
$file['unique_key0'] = My_General_Functions::gen_random_string(10);
$file['unique_key1'] = My_General_Functions::gen_random_string(10);

  $file_upload->setDestination(Zend_Registry::get('Config')->file_uploads->destination_dir);
$i = 0;
foreach($file_upload->getFileInfo() as $info) {
// get uploaded files extensions
$file_ext = $this->getExtension($info['name']);

  // generate file names for the ones to be stored on server
$fileName = 'file'.'_'.$i.'_'.$file["unique_key$i"].'.'.$file_ext;
$filename[$i] = $info['name'];

  $i++;

  // rename each of the uploaded files with random name based on unique_key0 and unique_key1 array
vars

  $file_upload->addFilter('Rename', array(
'target' => Zend_Registry::get('Config')->file_uploads->destination_dir . '/' . $fileName,
'overwrite' => 'yes'
));
if(!$file_upload->receive($info['name'])) {
$this->view->msg = $file_upload->getMessages();
return;
}
}

  public function getExtension ($name)
{
if($name){
$exts = preg_split("[/\\.]", $name);
```

```
$n = count($exts)-1;
$exts = $exts[$n];
return $exts;
}
```

In the above code example the **getExtension function** is prepared in order to extract the file extension, this is necessary to build up the uploaded file name.
In this code sample, the **foreach** loop will get through all the defined file names earlier on and will iterate through them setting each file a unique name in the format similar to **file_0_UVicsNlBcH.your_file.txt** .
As you can see the generated file has few parts **"file"** is a static one **0** is generated because the file to be saved is the first uploaded file from a defined Zend_Form. The **UVicsNlBcH** within the file name is an unique key by which you can later recognize the file name, this is especially handy if you also store the string somewhere in database.

The code:

```
  $file_upload->addFilter('Rename', array(
'target' => Zend_Registry::get('Config')->file_uploads->destination_dir . '/' . $fileName,
'overwrite' => 'yes'
));
```

is responsible for renaming each of the uploaded files to the custom generated file name. I believe the rest of the code is very easy to understand.
It took me a while until I come up with this code and I do not guarantee that this way of doing things is secure or anything it just seems to work for me for now so I decided to share it in an attempt to help to somebody developing a file uploader in Zend Framework.
The feedback or code improvement suggestions are very welcome!