

# [PHP] Classe Upload [1.0] | Documentation [FR]

## *Une classe simple pour gérer vos uploads*

### Licence

*Ce programme est libre, vous pouvez le redistribuer et/ou le modifier selon les termes de la Licence Publique Générale GNU publiée par la Free Software Foundation (version 2 ou bien toute autre version ultérieure choisie par vous).*

*Ce programme est distribué car potentiellement utile, mais SANS AUCUNE GARANTIE, ni explicite ni implicite, y compris les garanties de commercialisation ou d'adaptation dans un but spécifique. Reportez-vous à la Licence Publique Générale GNU pour plus de détails.*

### A propos

La classe Upload est une classe PHP vous permettant de gérer facilement les uploads de fichiers sur votre serveur. Elle vous permet de définir quelques règles simples de vérification, pour vous assurer que les fichiers envoyés correspondent à vos attentes. Vous pouvez l'utiliser comme bon vous semble en veillant cependant à respecter la licence GNU / GPL (voir ci-dessus).

### Introduction

La classe Upload est simple d'utilisation. Voici le code pour uploader un fichier de manière sommaire :

```
require('Upload.php');
$upload = new Upload('documents', $_FILES['file']);
$upload->upload();
```

Pour uploader votre fichier, il vous suffit d'inclure et d'instancier la classe Upload avec 2 paramètres.

1. Le premier est le dossier de destination du fichier à partir de la racine de votre site
2. Le second est le \$\_FILES récupéré

N'oubliez pas ! Pour recevoir un fichier via votre formulaire HTML, vous devez ajouter à votre balise <form> : `enctype='multipart/form-data'`

## Ajouter des règles/actions

Dans la version actuelle, la classe Upload permet l'ajout de 3 règles/actions :

1. Restriction des extensions autorisées
2. Restriction de taille de fichier
3. Application d'un nom personnalisé au fichier

### 1 ) Restreindre les extensions

Vous pouvez définir quelles extensions vous souhaitez autoriser pour le fichier. Par exemple, si vous ne souhaitez que des images, vous allez autoriser : png, jpg, gif.

Pour cela il vous suffit d'écrire la règle suivante une fois votre classe instanciée :

```
$upload->set_allowed_extensions(array('png', 'jpg', 'gif'));
```

La classe va automatiquement vérifier si le fichier envoyé est du bon type, et générer une erreur si ce n'est pas le cas.

L'utilisation des MIME Types étant désormais complètement obsolète en matière de sécurité, la performance a été préférée dans cette classe.

### 2 ) Restreindre la taille maximale

Vous pouvez définir une taille maximale (en octets) pour le fichier envoyé. Par exemple, si vous souhaitez la limiter à 2 Mo :

```
$upload-> set_max_size_allowed(200000);
```

La classe va automatiquement vérifier si le fichier envoyé a une taille inférieure (ou égale) à cette limite et générer une erreur si ce n'est pas le cas.

### 3 ) Définir un nom personnalisé

Vous pouvez définir le nom que portera le fichier sur votre serveur. Pour cela, il suffit d'utiliser `set_name()` :

```
$upload->set_name('Mon fichier de test !', true);
```

La seconde variable (false par défaut) permet de définir si le nom du fichier doit être réécrit, c'est-à-dire, s'il faut le formater pour supprimer les éventuels caractères spéciaux et espaces qu'il pourrait contenir. Si vous êtes certain du nom de votre fichier, vous pouvez simplement écrire :

```
$upload->set_name('mon-fichier-de-test');
```

## Gestion du résultat (erreurs / succès)

Le résultat de l'upload est retourné par la fonction upload. Vous l'obtenez donc en l'assignant à une variable :

```
$result = $upload->upload();
```

Ainsi, cette variable peut être :

- Un array contenant les erreurs
- Un boolean à TRUE si tout s'est bien déroulé
- Un boolean à FALSE (plus rarement) en cas d'erreur interne

On peut donc facilement traiter le résultat :

```
<?php if(is_array($result)) { ?>
    <h3>Erreur</h3>
    <ol>
        <?php foreach($result AS $k => $error) { ?>
            <li><?php echo $error; ?></li>
        <?php } ?>
    </ol>
<?php } else if($result === true) { ?>
    <p>Fichier envoyé !</p>
<?php } else { ?>
    <p>Erreur interne !</p>
<?php } ?>
```

## Accession aux variables

Voici les variables publiques accessibles depuis l'objet Upload :

- \$errors
- \$final\_name : Le nom final du fichier avec son extension (mon-fichier-de-test.jpg)

## Me contacter

Si vous rencontrez des difficultés sur l'utilisation de la classe Upload, si vous avez des idées pour la faire évoluer, si vous constatez la présence d'anomalies ou encore si vous souhaitez me remercier (sait-on jamais !) vous pouvez me contacter :

- Via mon site : <http://paulineghiazza.fr>
- Directement par mail : [pauline.ghiazza@gmail.com](mailto:pauline.ghiazza@gmail.com)