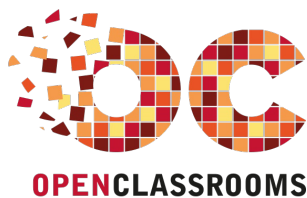


Page d'erreur perso

Par Kakyokuu

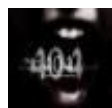


www.openclassrooms.com

*Licence Creative Commons 6 2.0
Dernière mise à jour le 13/10/2010*

Sommaire

Sommaire	2
Page d'erreur perso	3
Explications	3
S'assurer que les robots ne viendront pas	6
Méthode	6
Annexes	6
Annexe 1 : racine du site	7
Annexe 2 : robots.txt	7
Partager	7



Page d'erreur perso

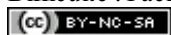


Par

Kakyokuu

Mise à jour : 13/10/2010

Difficulté : Facile



Vous en avez marre de voir les pages habituelles de Mozilla Firefox ? Opera ? Ou encore... Internet Explorer ? 🤔

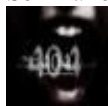
404 Not Found ? 500 Internal error ?

Grâce à ce tutoriel, vous pourrez personnaliser ces pages !



Cela risque cependant de ne pas fonctionner sur certains hébergeurs : si c'est le cas, faites-le-moi savoir.

Sommaire du tutoriel :



- [Explications](#)
- [S'assurer que les robots ne viendront pas](#)
- [Annexes](#)

Explications

Je vais vous expliquer ici comment créer une page d'erreur personnalisée, comme celle du Site du Zéro.

Rendez-vous par exemple sur <http://www.siteduzero.com/dkljd> : ça va vous afficher une erreur 404. Ou quand le site est saturé, c'est une erreur 500.

C'est un truc assez court, mais bien pratique !

Matériel nécessaire :

- des ciseaux ;
- un crayon ;
- de la colle ;
- une gomme ;
- un hébergeur acceptant le PHP et les .htaccess.

Créez maintenant un fichier .htaccess que vous placerez à la racine de votre site, et insérez-y ceci (en remplaçant bien sûr « <http://www.siteduzero.com> » par l'adresse de votre site web) :

Code : Apache

```
ErrorDocument 400 http://www.siteduzero.com/400.php
ErrorDocument 401 http://www.siteduzero.com/401.php
ErrorDocument 402 http://www.siteduzero.com/402.php
ErrorDocument 403 http://www.siteduzero.com/403.php
ErrorDocument 404 http://www.siteduzero.com/404.php
ErrorDocument 405 http://www.siteduzero.com/405.php
ErrorDocument 500 http://www.siteduzero.com/500.php
ErrorDocument 501 http://www.siteduzero.com/501.php
ErrorDocument 502 http://www.siteduzero.com/502.php
ErrorDocument 503 http://www.siteduzero.com/503.php
ErrorDocument 504 http://www.siteduzero.com/504.php
ErrorDocument 505 http://www.siteduzero.com/505.php
```



Mettez les liens en absolu pour l'instant, nous allons voir dans quel cas les utiliser en relatif.



C'est bien beau, les 400, 401, 402, 403, 404, 500... mais ça veut dire quoi ?



- 400 : échec de l'analyse HTTP.
- 401 : mauvais pseudo ou mot de passe dans le .htaccess.
- 402 : le client doit reformuler sa demande avec les bonnes données de paiement.
- 403 : requête interdite.
- 404 : page non trouvée.
- 405 : méthode non autorisée.
- 500 : erreur interne au serveur ou serveur saturé.
- 501 : le serveur ne supporte pas le service demandé.
- 502 : mauvaise passerelle.
- 503 : service indisponible.
- 504 : trop de temps à la réponse.
- 505 : version HTTP non supportée.

Maintenant, vous devez vous dire : « Je mets quoi dans 401.php ?... »

Vous y mettez ce que vous voulez.

Exemple pour 401.php : « Vous avez entré un mauvais pseudo ou mot de passe. »

Exemple pour 404.php : « Page non trouvée. »

Vous l'avez compris, les erreurs 400 sont des erreurs du client, et les 500 des erreurs du serveur ; je vous laisse mettre tout ça en forme ! 🤖



Le problème, c'est que ça va renvoyer vers une autre erreur : l'erreur 200 (page qui s'est correctement chargée). Je ne sais donc pas si on peut l'appeler une « erreur ».

Ainsi, avec les sites de référencement ayant un robot, comme Google ou MSN Search, il y a un léger problème : le moteur de recherche n'est pas en mesure de distinguer si la page est une page d'erreur ou non. Il affichera alors des liens vers des pages d'erreur lors d'une recherche.

Code : Apache

```
ErrorDocument 400 /400.php
ErrorDocument 401 /401.php
ErrorDocument 402 /402.php
ErrorDocument 403 /403.php
ErrorDocument 404 /404.php
ErrorDocument 405 /405.php
ErrorDocument 500 /500.php
ErrorDocument 501 /501.php
ErrorDocument 502 /502.php
ErrorDocument 503 /503.php
ErrorDocument 504 /504.php
ErrorDocument 505 /505.php
```

Vous pouvez également écrire un petit message sans créer de nouvelle page, comme ceci :

Code : Apache

```
ErrorDocument 400 "Échec de l'analyse HTTP."
ErrorDocument 401 "Mauvais pseudo ou mot de passe."
ErrorDocument 402 "Le client doit reformuler sa demande avec les
bonnes données de paiement."
ErrorDocument 403 "..."
ErrorDocument 404 ""
```

```
ErrorDocument 405 ""
ErrorDocument 500 ""
ErrorDocument 501 ""
ErrorDocument 502 ""
ErrorDocument 503 ""
ErrorDocument 504 ""
ErrorDocument 505 ""
```

Vous l'aurez compris : vous devez inscrire le message d'erreur entre les guillemets.

Mais il est énervant de devoir créer une nouvelle page à chaque fois ! Eh bien, grâce à du PHP, vous pouvez n'en créer qu'une contenant toutes les erreurs ; regardez :

Code : Apache

```
ErrorDocument 400 /erreur.php?erreur=400
ErrorDocument 401 /erreur.php?erreur=401
ErrorDocument 402 /erreur.php?erreur=402
ErrorDocument 403 /erreur.php?erreur=403
ErrorDocument 404 /erreur.php?erreur=404
ErrorDocument 405 /erreur.php?erreur=405
ErrorDocument 500 /erreur.php?erreur=500
ErrorDocument 501 /erreur.php?erreur=501
ErrorDocument 502 /erreur.php?erreur=502
ErrorDocument 503 /erreur.php?erreur=503
ErrorDocument 504 /erreur.php?erreur=504
ErrorDocument 505 /erreur.php?erreur=505
```

On va ensuite savoir quelle erreur afficher dans la page erreur.php grâce à la valeur du GET *erreur*.

Code : PHP

```
<?php
switch($_GET['erreur'])
{
    case '400':
        echo 'Échec de l\'analyse HTTP.';
        break;
    case '401':
        echo 'Le pseudo ou le mot de passe n\'est pas correct !';
        break;
    case '402':
        echo 'Le client doit reformuler sa demande avec les bonnes
données de paiement.';
        break;
    case '403':
        echo 'Requête interdite !';
        break;
    case '404':
        echo 'La page n\'existe pas ou plus !';
        break;
    case '405':
        echo 'Méthode non autorisée.';
        break;
    case '500':
        echo 'Erreur interne au serveur ou serveur saturé.';
        break;
    case '501':
        echo 'Le serveur ne supporte pas le service demandé.';
        break;
    case '502':
        echo 'Mauvaise passerelle.';
        break;
    case '503':
        echo 'Service indisponible.';
        break;
}
```

```
break;
case '504':
echo 'Trop de temps à la réponse.';
break;
case '505':
echo 'Version HTTP non supportée.';
break;
default:
echo 'Erreur !';
}
?>
```

S'assurer que les robots ne viendront pas

Comme je l'ai dit plus haut, il y a un risque que vos pages d'erreur soient référencées dans les moteurs de recherche, nous allons donc leur interdire l'accès.

Méthode

Nous allons implanter un fichier robots.txt à la racine de notre site.
Il aura pour but d'indiquer à tel ou tel robot de ne pas visionner telle ou telle page.

Voici comment se présente ce fichier :

Code : Autre - robots.txt

```
User-Agent:
Disallow:
```

À la suite de **User-Agent**, vous inscrirez la liste des robots dont vous souhaitez interdire l'accès. Nous allons toutefois utiliser ici l'étoile (*), qui va interdire l'accès à tous les robots.

Et après **Disallow**, vous indiquerez la liste des pages à bloquer, les unes après les autres, en réécrivant « Disallow » pour chaque page :

Code : Autre - robots.txt

```
User-Agent: *
Disallow: /400.php
Disallow: /401.php
Disallow: /402.php
Disallow: /403.php
```



Stop ! Je ne vais pas toutes les mettre alors qu'il existe une trentaine d'erreurs, si ?

Non, vous pouvez interdire l'accès à tout un dossier. Je vous conseille donc vivement d'enregistrer ces pages dans un dossier à part et de ne pas oublier de changer l'URI ou l'URL (c'est-à-dire l'adresse) que vous avez mise dans le fichier .htaccess. Pour interdire l'accès à un dossier, pas de mystère, cela se passe comme pour les fichiers.

Code : Autre - robots.txt

```
User-Agent: *
Disallow: /nom_du_dossier_ou_se_trouvent_les_erreurs/
```

C'est tout de suite plus lisible, n'est-ce pas ?

Annexes

Ces annexes vont vous permettre de comprendre quelques petites choses qui auraient pu être incomprises ou mal comprises.

Annexe 1 : racine du site

Explication

La racine du site est la base de la hiérarchie de votre site, c'est là où le visiteur arrive lorsqu'il entre l'adresse du site dans la barre d'adresse de son navigateur (ce qui suppose qu'il n'y ait au préalable aucune redirection).

Exemples d'arborescences

Si vous possédez un hébergement mutualisé, vous n'aurez accès qu'à une partie restreinte du serveur. Par exemple, sous Linux, vous pourriez avoir accès à la partie /home/www/heberg_id_674123/.

Seule la partie après heberg_id_674123 vous sera visible, aucun accès ne vous sera autorisé dans un dossier parent.

Les hébergeurs ont plusieurs solutions : soit il mettent le dossier heberg_id_674123 comme dossier racine du site, soit ils vont vous permettre de ne pas mettre de fichier dans la partie publique mais dans une partie privée.

Exemples :

```
/home/www/ [...]
...../heberg_id_674121/[...]
...../heberg_id_674122/[...]
...../heberg_id_674123/private
...../public ← Racine du site
```

```
/home/www/ [...]
...../heberg_id_674121/[...]
...../heberg_id_674122/[...]
...../heberg_id_674123/
...../www ← Racine du site
```

```
/home/www/ [...]
...../heberg_id_674121/[...]
...../heberg_id_674122/[...]
...../heberg_id_674123/ ← Racine du site
```

Annexe 2 : robots.txt

Le fichier robots.txt est un fichier présent à la racine du site qui contient des règles pour les robots (de Google, Yahoo, etc.) qui viendraient sur votre site. Ce fichier est vérifié et respecté par tous les robots dignes de ce nom (donc les plus connus), mais beaucoup ne respectent pas ces règles, voire ne les lisent même pas. Ne vous faites donc pas trop d'illusions concernant ce fichier : c'est respecté par une petite poignée de robots, mais pas par tous.

Et voilà : ce n'est pas plus beau ainsi ?

Bien entendu, vous pouvez créer votre propre mise en page, avec du CSS par exemple ! Mais bon... que ça ne devienne pas une usine à gaz. 🤖

Merci à tous les Zéros pour l'aide et le soutien qu'ils m'ont apportés dans l'élaboration de ce tutoriel.

Veuillez s'il vous plaît me signaler tout problème par MP.

Partager



Ce tutoriel a été corrigé par les [zCorrecteurs](#).