

Combo - Development #48413

assets: servir les fichiers du répertoire media si "key" commence par "media:"

09 novembre 2020 17:04 - Serghei Mihai

Statut:	Rejeté	Début:	09 novembre 2020
Priorité:	Normal	Echéance:	
Assigné à:	Serghei Mihai	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
Pour un appel GET /assets/media:image chercher le fichier image dans le répertoire media et retourner son URL.			
Demandes liées:			
Lié à Intégrations graphiques Publik - Development #48415: arles-2020: rajout...			Fermé 09 novembre 2020

Historique

#2 - 09 novembre 2020 18:00 - Serghei Mihai

- Fichier 0001-assets-allow-ckeditor-s-media-files-serving-46262.patch ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

#3 - 09 novembre 2020 18:21 - Frédéric Péters

```
+ if key.startswith('media:')
```

On pourrait imaginer tout ça sous une URL séparée, pour s'épargner ce test.

```
+ if not asset.is_image():
```

Pourquoi accepter uniquement les images et pas les autres types de fichiers, ça me semble arbitraire ?

```
+ root, ext = os.path.splitext(asset.name)
```

Je ne suis vraiment pas fan d'appeler ça "root" (même si c'est la forme reprise dans la documentation Python), tu pourrais juste zapper la recherche d'un nom convenable,

```
if os.path.splitext(asset.name)[0] == ...:
```

Plutôt faire le key.replace() en-dehors de la boucle. (et ajouter , 1 derrière pour juster virer l'occurrence du début).

Aujourd'hui,

```
url(r'^assets/(?P<key>[\w_-]+)$', views.serve_asset),
```

ça viendrait poser certaines exigences sur les caractères du fichier.

Mais en fait ce ticket c'est 1/ pour ne pas avoir à donner d'extension (mais ça me semble accessoire), 2/ contourner le fait que les fichiers des médias sont uploadés dans des répertoires datés. C'est bien ça ?

J'ai l'impression que oui, et que le test est à la ramasse,

```
+ Asset(key='media:test', asset=File(BytesIO(b'test'), 'test.png')).save()
```

qu'il s'agit justement de traiter des fichiers où il n'y a pas d'objet Asset.

Au final c'est peu de code et ça me laisserait aller à accepter (surtout si c'était tapé sur une URL différente), mais quand même, ça me semble une approche totalement ad hoc, sans perspective.

#4 - 09 novembre 2020 20:32 - Serghei Mihai

- Fichier 0001-assets-add-ckeditor-s-media-files-serving-46262.patch ajouté

Frédéric Péters a écrit :

[...]

On pourrait imaginer tout ça sous une URL séparée, pour s'épargner ce test.

J'ai créé une nouvelle vue et url.

Pourquoi accepter uniquement les images et pas les autres types de fichiers, ça me semble arbitraire ?

Je suis parti du principe qu'on chercherait uniquement des images, mais ok pour zapper la vérification.

Au final c'est peu de code et ça me laisserait aller à accepter (surtout si c'était tapé sur une URL différente), mais quand même, ça me semble une approche totalement ad hoc, sans perspective.

Oui, c'est du dev ad-hoc sans énormément d'intérêt.

#5 - 09 novembre 2020 20:45 - Frédéric Péters

Oui, c'est du dev ad-hoc (...)

Ce qui généralement n'aurait pas sa place ici.

Je posais cette question :

Mais en fait ce ticket c'est 1/ pour ne pas avoir à donner d'extension (mais ça me semble accessoire), 2/ contourner le fait que les fichiers des médias sont uploadés dans des répertoires datés. C'est bien ça ?

Je la posais pour ne pas partir à réfléchir à des changements qui répondraient à ces questions si ce n'étaient pas les bonnes.

~

Le patch attaché modifie encore la vue "serve_asset".

Aussi, je ne l'écrivais pas, mais une raison pour passer à une vue particulière serait de dégager la contrainte `[w_]+`, que tu gardes, et qui fait partie du côté inutilement ad hoc du patch.

Pour aller jusqu'au bout, je dirais qu'on veut `assets/media-lookup/(?P<filename>[w_]+)`.

Dans `def serve_media_asset(request, key)`: `key` serait remplacé par `filename`; et dans la boucle ça matcherait aussi si le nom complet du fichier correspond; dans cette idée :

```
- def serve_media_asset(request, key):
+ def serve_media_asset(request, filename):
    for asset in CkEditorAsset.get_assets(request):
-         if os.path.splitext(asset.name)[0] == key:
+         if asset.name == filename os.path.splitext(asset.name)[0] == filename:
```

Ainsi déjà on irait vers quelque chose qu'on pourrait considérer plus généralement utile.

#6 - 10 novembre 2020 09:50 - Serghei Mihai

- Fichier `0001-assets-add-ckeditor-s-media-files-serving-46262.patch` ajouté

Yep, rechercher le fichier par son nom complet fait plus de sens.

#7 - 10 novembre 2020 09:52 - Frédéric Péters

Pourquoi encore exclure certains caractères (autres que le `/`) ?

#8 - 10 novembre 2020 11:01 - Serghei Mihai

Frédéric Péters a écrit :

Pourquoi encore exclure certains caractères (autres que le `/`) ?

Branche à jour avec uniquement l'exclusion de `/` dans la regex.

#9 - 13 novembre 2020 10:26 - Serghei Mihai

- Lié à *Development #48415: arles-2020: rajouter le template et styles pour le champ liste personnalisé ajouté*

#10 - 17 novembre 2020 14:20 - Serghei Mihai

- Fichier *0001-assets-add-ckeditor-s-media-files-serving-46262.patch* ajouté

Nouveau patch de la branche à jour.

#11 - 11 décembre 2020 15:03 - Lauréline Guérin

- Assigné à mis à *Serghei Mihai*

#12 - 05 janvier 2021 13:40 - Frédéric Péters

C'est une fonctionnalité encore souhaitée ?

#13 - 05 janvier 2021 13:56 - Serghei Mihai

- Statut *changé de Solution proposée à Rejeté*

Non, je rejette.

Fichiers

0001-assets-allow-ckeditor-s-media-files-serving-46262.patch	2,15 ko	09 novembre 2020	Serghei Mihai
0001-assets-add-ckeditor-s-media-files-serving-46262.patch	3,01 ko	09 novembre 2020	Serghei Mihai
0001-assets-add-ckeditor-s-media-files-serving-46262.patch	2,62 ko	10 novembre 2020	Serghei Mihai
0001-assets-add-ckeditor-s-media-files-serving-46262.patch	2,62 ko	17 novembre 2020	Serghei Mihai