

Intégrations graphiques Publik - Bug #84816

Fichiers manquant quand on lint vaguement les fichiers style.css

17 décembre 2023 01:08 - Benjamin Dauvergne

Statut:	Nouveau	Début:	17 décembre 2023
Priorité:	Bas	Echéance:	
Assigné à:		% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposé:	Non		

Description

En tentant d'amener notre infra de thème à utiliser ManifestStaticFileStorage pour obtenir des URLs hashés pour les feuilles de style et leur dépendance j'ai fait crasher collectstatic qui n'arrive pas à trouver certains fichiers référencés, pour la plupart ce sont des versions .woff des polices que nous ne distribuons pas mais qui sont quand même référencées comme alternatives, mais il y a aussi des fichiers images qui je pense devraient être référencés correctement mais ne le sont pas car les imports .scss ne corrigent pas les URLs relatives pour les images, exemples sur amiens-metropole:

```
# amien-metropole/style.scss
...
@import '../includes/categories';
```

dans includes/_categories.scss

```
&.#{$category} h2:first-child { background-image: url(img/#{$category}.png); }
```

ce qui donne par exemple ces lignes dans amiens-metropole/style.css :

```
.gru-content div.cell.wcsformsofcategorycell.dechets.toggled h2:first-child {
  background-image: url(img/dechets-on.png); }
```

sauf que amiens-metropole/img/dechets-on.png n'existe pas.

Je ne sais pas trop si c'est grave concernant les thèmes, si ça n'a pas été remarqué ou si c'est connu et ça n'a aucun impact mais c'est embêtant pour éventuellement utiliser ManifestStaticFileStorage (j'ai développé en parallèle un ThemesDirectoryFinder pour pouvoir collecter les thèmes avec l'infra classique Django) ou même utiliser un linter sur les fichiers .css à un moment.

J'attache le rapport complet des fichiers référencés qui n'existent pas, en rangement de premier niveau le fichier .css et ensuite la liste des chemins relatifs qui ne sont pas trouvés par collectstatic quand il essaie de réécrire le fichier avec des liens contenant le hash du fichier cible.

Il y a aussi des soucis avec xstatic-roboto-fontface mais ce sont les histoires de fichiers .woff absent qui ne sont pas très graves et seront simples à corriger et aussi des faux positifs dans jquery-ui parce que la recherche par regexp de Django n'est pas capable d'ignorer les fichiers référencés au milieu d'un commentaire.

PS: un vieux ticket sass sur le sujet <https://github.com/sass/sass/issues/2701>

Demandes liées:

Lié à Intégrations graphiques Publik - Development #32660: utiliser des URL a...

Fermé

28 avril 2019

Historique

#1 - 17 décembre 2023 17:20 - Benjamin Dauvergne

- Description mis à jour

#2 - 17 décembre 2023 17:26 - Benjamin Dauvergne

J'ai l'impression qu'une discipline de rangement et de nommage suffirait :

- tous les assets soit dans <theme>/{fonts,img}/asset.ext ou {fonts,img}/asset.ext
- toutes les feuilles de style dans <theme>/xxx.{css,scss}

- tous les chemins de la forme ../<theme>/{fonts,img}/asset.ext ou ../{fonts,img}/asset.ext

Dans ce cas tous les chemins même importés seront résolus. Je vois aussi pas mal de chemins absolus ('/static/...') intégrant le préfixe utilisé lors du déploiement, ce serait aussi une solution, mais ça fait des chemins plus long.

#3 - 17 décembre 2023 17:58 - Benjamin Dauvergne

- *Description mis à jour*

#4 - 17 décembre 2023 17:58 - Benjamin Dauvergne

- *Lié à Development #32660: utiliser des URL absolues pour les /static/ ajouté*

Fichiers

report.txt

109 ko 17 décembre 2023

Benjamin Dauvergne