

Authentic 2 - Bug #13587

include all attributes in user export

13 octobre 2016 14:32 - Frédéric Péters

Statut:	Fermé	Début:	13 octobre 2016
Priorité:	Normal	Echéance:	
Assigné à:		% réalisé:	100%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	
Patch proposed:	Oui		
Description			
From the page listing members of a role it is possible to ask for a CSV export but it doesn't include all the fields; it should.			

Révisions associées

Révision 0bf7fe9f - 02 mai 2018 19:44 - Benjamin Dauvergne

manager: clean resources file (#13587)

Remove unused imports, remove commented code.

Révision 89643033 - 02 mai 2018 19:44 - Benjamin Dauvergne

manager: allow overriding resource class in export views (#13587)

Révision 516cb4f8 - 02 mai 2018 19:44 - Benjamin Dauvergne

utils: add a batch_queryset method to load large prefetched queryset without exhausting memory (#13587)

Révision cc101ea3 - 02 mai 2018 19:44 - Benjamin Dauvergne

manager: add a default implementation of ExportMixin.get_data() (#13587)

It takes the default queryset and batches it using the new function batch_queryset().

Révision 7a6ce74d - 02 mai 2018 19:44 - Benjamin Dauvergne

tests: add test of CSV export of users (#13587)

Révision 19b00d8a - 02 mai 2018 19:44 - Benjamin Dauvergne

manager: export all user attributes (fixes #13587)

Révision df942ea1 - 02 mai 2018 19:44 - Benjamin Dauvergne

custom_user: cache attribute list during a request (#13587)

Historique

#1 - 24 novembre 2016 14:19 - Benjamin Dauvergne

~~Would authentic2.utils.login() be a good placement for this callback?~~ (reposted in [#14070](#))

#2 - 25 avril 2018 16:42 - Benjamin Dauvergne

- Fichier 0001-manager-clean-resources-file-13587.patch ajouté
- Fichier 0002-manager-allow-overriding-resource-class-in-export-vi.patch ajouté
- Fichier 0003-manager-export-all-user-attributes-fixes-13587.patch ajouté
- Fichier 0004-custom_user-cache-attribute-list-during-a-request-13.patch ajouté
- Patch proposed changé de Non à Oui

- 0001: nettoyage
- 0002: permet de customiser l'objet Resource dans la vue d'export
- 0003: ajout de l'export des attributs supplémentaires des utilisateurs

- 0004: optimisation pour ne pas faire un SELECT * FROM authentic2_attribute pour chaque utilisateur

#3 - 02 mai 2018 14:04 - Emmanuel Cazenave

Je peux pas acker si ya pas un test.

Deux import inutiles dans 0002-manager-allow-overriding-resource-class-in-export-vi.patch.

Et aussi, vieux débat, mais pas pour ma part pas fan de l'utilisation qui est faite de l'héritage : avec méthode 'finale' get_dataset dans la classe mère, dont le comportement est modifié dans une classe fille parce qu'on y redéfinit get_resource.
Je trouverais ça plus clair si c'était get_dataset qui était redéfinie.

#4 - 02 mai 2018 15:15 - Benjamin Dauvergne

Emmanuel Cazenave a écrit :

Je peux pas acker si ya pas un test.

Ok.

Deux import inutiles dans 0002-manager-allow-overriding-resource-class-in-export-vi.patch.

Ok.

Et aussi, vieux débat, mais pas pour ma part pas fan de l'utilisation qui est faite de l'héritage : avec méthode 'finale' get_dataset dans la classe mère, dont le comportement est modifié dans une classe fille parce qu'on y redéfinit get_resource.
Je trouverais ça plus clair si c'était get_dataset qui était redéfinie.

J'ai l'impression de suivre le modèle générale des vues générique que ce soit dans Django(get_form_class, get_form_kwargs, get_form) ou dans DRF (pareil avec get_serializer, etc...). Dès qu'un objet un peu utilisé partout dans la vue est construit on déplace le constructeur dans une méthode à part pour pouvoir le customiser. Là tout de suite je ne vois pas le besoin de changer quelque chose.

La surcharge actuelle de get_dataset() sert uniquement à contourner un problème avec django-import-export qui annule les effets du prefetching quand on passe un queryset (voir <https://github.com/django-import-export/django-import-export/issues/774>).

Je pense que je vais remonter ça au niveau de classe mère parce que ça n'a rien de spécifique.

#5 - 02 mai 2018 16:48 - Benjamin Dauvergne

- Fichier 0005-tests-add-test-of-CSV-export-of-users-13587.patch ajouté
- Fichier 0001-manager-clean-resources-file-13587.patch ajouté
- Fichier 0002-manager-allow-overriding-resource-class-in-export-vi.patch ajouté
- Fichier 0007-custom_user-cache-attribute-list-during-a-request-13.patch ajouté
- Fichier 0003-utils-add-a-batch_queryset-method-to-load-large-pref.patch ajouté
- Fichier 0004-manager-add-a-default-implementation-of-ExportMixin..patch ajouté
- Fichier 0006-manager-export-all-user-attributes-fixes-13587.patch ajouté

3 patches en plus et révision du patch 'export all user attributes':

- 0003: ajout d'une fonction authentic2.utils.batch_queryset() pour aider à traverser un gros queryset avec du prefetch sans tout charger
- 0004: utilisation de cette nouvelle fonction pour donner une implémentation par défaut de get_data() à ExportMixin
- 0005: ajout d'un test avec mesure exacte du nombre de requêtes
- 0006: maintenant c'est bien séparé, dans get_resource() on travaille sur la classe ressource dans get_queryset() on ajoute du prefetch, le get_data() entre les deux est invisible, c'est la tambouille de ExportMixin.

#6 - 02 mai 2018 17:53 - Emmanuel Cazenave

Toujours les deux imports inutiles qui traînent.

J'essayais de comprendre à coup de pdb ce qui allait se passer sur qs = super(UsersExportView, self).get_queryset() parce qu'avais l'impression qu'aucune classe parente ne définit cette méthode, mais j'ai pas pu arriver jusque là, le test plante.

```
NameError: global name 'superuser' is not defined
```

Ma remarque sur l'héritage, de façon plus formelle c'est ça : <https://codeburst.io/understanding-solid-principles-open-closed-principle-e2b588b6491f>
Mais d'une part tout le monde est pas d'accord, d'autre part sur ton code c'était suffisamment petit pour que ce soit pas un gros problème.
Mais quand ça devient plus gros, genre un héritage à 3 niveaux et des redéfinition de certaines méthodes dans les classes filles qui seront appelées

par les classe mères, ça peut vite devenir tout à fait incompréhensible.

Tout ça pour dire fais comme tu veux.

#7 - 02 mai 2018 18:49 - Benjamin Dauvergne

Corrigé les deux soucis, je vais pousser quand mes tests ont fini de tourner.

Pour le reste, j'aime bien ces idées (préférer la composition à l'héritage, has-a plutôt que is-a), mais Django n'est pas du tout construit en suivant ce principe, c'est de l'héritage partout avec des bouts de composition (tous les systèmes avec des backends), je préfère la cohérence avec notre framework de base plutôt que d'avoir à empiler des styles de code différents (je pense que ce serait encore pire). Ça veut dire qu'un regarde extérieur habitué à du code Django (ou regardant des exemples dans les docs) ne sera pas trop perdu.

#8 - 02 mai 2018 19:45 - Benjamin Dauvergne

- Statut changé de Nouveau à Résolu (à déployer)

- % réalisé changé de 0 à 100

Appliqué par commit [authentic2|19b00d8ac55d070b84fbc81db5340944cac70d11](#).

#9 - 13 décembre 2018 22:31 - Benjamin Dauvergne

- Statut changé de Résolu (à déployer) à Fermé

Fichiers

0001-manager-clean-resources-file-13587.patch	1,8 ko	25 avril 2018	Benjamin Dauvergne
0002-manager-allow-overriding-resource-class-in-export-vi.patch	1,5 ko	25 avril 2018	Benjamin Dauvergne
0003-manager-export-all-user-attributes-fixes-13587.patch	1,86 ko	25 avril 2018	Benjamin Dauvergne
0004-custom_user-cache-attribute-list-during-a-request-13.patch	1,87 ko	25 avril 2018	Benjamin Dauvergne
0005-tests-add-test-of-CSV-export-of-users-13587.patch	2,59 ko	02 mai 2018	Benjamin Dauvergne
0001-manager-clean-resources-file-13587.patch	1,8 ko	02 mai 2018	Benjamin Dauvergne
0002-manager-allow-overriding-resource-class-in-export-vi.patch	1,5 ko	02 mai 2018	Benjamin Dauvergne
0007-custom_user-cache-attribute-list-during-a-request-13.patch	1,87 ko	02 mai 2018	Benjamin Dauvergne
0003-utils-add-a-batch_queryset-method-to-load-large-pref.patch	1,65 ko	02 mai 2018	Benjamin Dauvergne
0004-manager-add-a-default-implementation-of-ExportMixin..patch	3,03 ko	02 mai 2018	Benjamin Dauvergne
0006-manager-export-all-user-attributes-fixes-13587.patch	1,93 ko	02 mai 2018	Benjamin Dauvergne