Passerelle - Development #17163

json data store : fournir creation_datetime / last_update_datetime dans les résultats.

23 juin 2017 17:25 - Frédéric Péters

Statut:	Fermé	**Début:**	23 juin 2017
Priorité:	Normal	**Echéance:**	
Assigné à:	Valentin Deniaud	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Oui	**Planning:**	Non

Description

-

Historique

#1 - 02 octobre 2019 15:00 - Valentin Deniaud

- Fichier 0001-jsondatastore-return-datetimes-in-results.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Assigné à mis à Valentin Deniaud

- Patch proposed changé de Non à Oui

Est-ce qu'il y a besoin d'un test ? Est-ce que le format par défaut passe ou il vaut mieux mettre un coup de strftime pour que ce soit plus joli ?

#2 - 02 octobre 2019 15:08 - Frédéric Péters

Oui pour le test.

Pour les datetimes, ils devraient être sérialisés automatiquement, pas besoin de str().

#3 - 02 octobre 2019 17:44 - Valentin Deniaud

- Fichier 0001-jsondatastore-return-datetimes-in-results.patch ajouté

J'ai enlevé les str et j'ai rajouté un test. Il a une bizarrerie parce que django rest framework ne formate pas les dates comme isoformat de datetime : il tronque les ms (==> ne pas en mettre) et il met des Z aux timezone UTC (==> enlever les Z...).

#4 - 02 octobre 2019 17:48 - Frédéric Péters

> django rest framework

La sérialisation devrait être celle de JSONEncoder dans passerelle/utils/jsonresponse.py, curieux d'avoir un Z qui apparait.

#5 - 02 octobre 2019 17:50 - Frédéric Péters

Mais plutôt que faire ce mock, tu peux juste comparer avec l'objet créé dans la base.

#6 - 02 octobre 2019 17:53 - Frédéric Péters

> La sérialisation devrait être celle de JSONEncoder dans passerelle/utils/jsonresponse.py, curieux d'avoir un Z qui apparait.

Sauf que là ça prend juste le time.struct_time, je n'ai rien dit.

#7 - 02 octobre 2019 17:56 - Frédéric Péters

Mon deuxième commentaire, comparer l'objet plutôt que mocker() un faux datetime, par contre, reste valable.

#8 - 02 octobre 2019 18:07 - Valentin Deniaud

En fait c'est Django tout court qui formate comme ça, pas DRF https://docs.djangoproject.com/en/2.2/topics/serialization/#djangojsonencoder. Mais je vais faire comme tu dis, c'est clairement mieux.

#9 - 03 octobre 2019 10:27 - Valentin Deniaud

- Fichier 0001-jsondatastore-return-datetimes-in-results.patch ajouté

Voilà.

#10 - 03 octobre 2019 10:53 - Benjamin Dauvergne

- Statut changé de Solution proposée à Solution validée

#11 - 03 octobre 2019 11:15 - Valentin Deniaud

- Statut changé de Solution validée à Résolu (à déployer)

```
commit eb1c4ad699d5ff5dd26bd58312366beee6cd1c0d
Author: Valentin Deniaud <vdeniaud@entrouvert.com>
Date:   Wed Oct 2 14:54:55 2019 +0200

    jsondatastore: return datetimes in results
```

#12 - 09 décembre 2019 14:48 - Valentin Deniaud

- Statut changé de Résolu (à déployer) à Solution déployée

C'est en prod depuis longtemps.

Fichiers

0001-jsondatastore-return-datetimes-in-results.patch	2,27 ko	02 octobre 2019	Valentin Deniaud
0001-jsondatastore-return-datetimes-in-results.patch	4,04 ko	02 octobre 2019	Valentin Deniaud
0001-jsondatastore-return-datetimes-in-results.patch	4,09 ko	03 octobre 2019	Valentin Deniaud