Chrono - Bug #60351

API - CommaSeparatedStringField & entier

06 janvier 2022 14:42 - Lauréline Guérin

Statut:	Fermé	**Début:**	06 janvier 2022
Priorité:	Normal	**Echéance:**	
Assigné à:	Nicolas Roche	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Oui	**Planning:**	Non

Description

Si on reçoit un entier, ça pète:

```
chrono ERROR agendas-venissieux-test.demarches.sitiv.fr 192.168.66.24 - r:7F21049FA518 Internal Se
rver Error: /api/agenda/epj-charreard-max-barel-atelier-sociocultur
                                                    Traceback (most recent call last):
                                                      File "/usr/lib/python3/dist-packages/django/core
/handlers/exception.py", line 34, in inner
                                                        response = get_response(request)
                                                      File "/usr/lib/python3/dist-packages/django/core
/handlers/base.py", line 115, in _get_response
                                                        response = self.process_exception_by_middlewar
e(e, request)
                                                      File "/usr/lib/python3/dist-packages/django/core
/handlers/base.py", line 113, in _get_response
                                                        response = wrapped_callback(request, *callback
_args, **callback_kwargs)
                                                      File "/usr/lib/python3/dist-packages/django/view
s/decorators/csrf.py", line 54, in wrapped_view
                                                        return view_func(*args, **kwargs)
                                                      File "/usr/lib/python3/dist-packages/django/view
s/generic/base.py", line 71, in view
                                                        return self.dispatch(request, *args, **kwargs)
                                                      File "/usr/lib/python3/dist-packages/rest_framew
ork/views.py", line 495, in dispatch
                                                        response = self.handle_exception(exc)
                                                      File "/usr/lib/python3/dist-packages/rest_framew
ork/views.py", line 455, in handle_exception
                                                        self.raise_uncaught_exception(exc)
                                                      File "/usr/lib/python3/dist-packages/rest_framew
ork/views.py", line 492, in dispatch
                                                        response = handler(request, *args, **kwargs)
                                                      File "/usr/lib/python3/dist-packages/chrono/api/
views.py", line 2163, in post
                                                        if not serializer.is_valid():
                                                      File "/usr/lib/python3/dist-packages/rest_framew
ork/serializers.py", line 236, in is_valid
                                                        self._validated_data = self.run_validation(sel
f.initial_data)
                                                      File "/usr/lib/python3/dist-packages/rest_framew
ork/serializers.py", line 434, in run_validation
                                                        value = self.to_internal_value(data)
                                                      File "/usr/lib/python3/dist-packages/rest_framew
ork/serializers.py", line 488, in to_internal_value
                                                        validated_value = field.run_validation(primiti
ve_value)
                                                      File "/usr/lib/python3/dist-packages/rest_framew
ork/fields.py", line 536, in run_validation
                                                        value = self.to_internal_value(data)
                                                      File "/usr/lib/python3/dist-packages/chrono/api/
serializers.py", line 33, in to_internal_value
```

```
                                                    data = [s.strip() for s in data.split(',') if
s.strip()]
                                                AttributeError: 'int' object has no attribute 'spl
it'
```

Révisions associées

Révision e532eefc - 15 février 2022 11:52 - Nicolas Roche

tests: change post/patch to pass json to event view (#60351)

Révision ac41d09c - 15 février 2022 11:52 - Nicolas Roche

api: use StringOrListField for recurrence_days serialiser (#60351)

Historique

#2 - 06 janvier 2022 14:52 - Valentin Deniaud

Note, plutôt que de chercher à réparer CommaSeparatedStringField, il faut le remplacer par StringOrListField et ensuite réparer si ça ne marche toujours pas.

La raison c'est #59813, w.c.s. est capable de passer les données POST sous forme de liste, donc il faut permettre d'exploiter ça. CommaSeparatedStringField devrait être réservé aux listes passées en paramètre d'URL, donc toujours des chaînes de caractères et jamais des vraies listes.

#3 - 06 janvier 2022 14:55 - Lauréline Guérin

(note: c'était sur le endpoint de création d'event)

#4 - 20 janvier 2022 18:27 - Nicolas Roche

- Assigné à mis à Nicolas Roche

> (note: c'était sur le endpoint de création d'event)

oups, c'est pour moi

dans le ticket lié

> le paramètre recurrence_days ... renvoyait un entier

Pour l'instant je n'arrive pas à reproduire via les tests.

#5 - 27 janvier 2022 17:07 - Nicolas Roche

- Fichier 0002-api-improve-recurrence_days-serialiser-on-create-eve.patch ajouté

- Fichier 0001-test-add-a-recurent-event-providing-int-as-recurrent.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

Bug reproduit en utilisant une expression python via wcs, ex: int(2).

0001: Pour être sûr que le pasage à StringOrListField ne casse rien.
0002: Remplacer par StringOrListField et ensuite réparer.

#6 - 31 janvier 2022 13:47 - Valentin Deniaud

Je comprends pas

```
        if isinstance(data, list) and len(data) == 1:
            data = data[0]
```

dans 0002 ?

#7 - 02 février 2022 17:17 - Nicolas Roche

L'idée c'est de continuer à gérer les "CommaSeparatedStringField" dans le champs POST,

pour éviter la régression sur ce test : tests/api/test_event.py::test_add_event_serialiser

```
        params = {
            'start_datetime': '2022-01-27 16:00',
            'places': 12,
            'recurrence_days': '0, 3, 5',
        }
        resp = app.post(api_url, params=params)
```

le serializer reçoit la liste ['0, 3, 5'], et il passe ses éléments
au validateur enfant (serializers.IntegerField) qui reçoit la chaîne '0, 3, 5'.

```
class StringOrListField(serializers.ListField):
    def to_internal_value(self, data):
        data_initial = data
        if isinstance(data, list) and len(data) == 1:
            data = data[0]
        if isinstance(data, str):
            data = [s.strip() for s in data.split(',') if s.strip()]
        import pdb; pdb.set_trace()
>       pasglop = super().to_internal_value(data_initial)
        return super().to_internal_value(data)
```

```
(Pdb) data_initial
['0, 3, 5']
(Pdb) n
rest_framework.exceptions.ValidationError: {0: [ErrorDetail(string='A valid integer is required.', code='inval
id')]}
(Pdb) d
(Pdb) d
rest_framework/fields.py(1688)run_child_validation()
(Pdb) self.child.run_validation('0, 3, 5')
*** rest_framework.exceptions.ValidationError: [ErrorDetail(string='A valid integer is required.', code='inval
id')]
```

(je n'ai pas compris pourquoi l'on reçoit une liste ici mais j'avoue que je n'ai pas encore testé pour voir ce que l'on reçoit de wcs et si ce sont en fait ce ne serait pas plutôt les tests qui sont faux)

#8 - 02 février 2022 17:33 - Nicolas Roche

- Statut changé de Solution proposée à En cours

Sauf que ça crashera si wcs envoie une (vraie) liste avec 1 seul entier.

#9 - 02 février 2022 19:11 - Nicolas Roche

- Fichier 0002-api-accept-int-on-recurrence_days-serialiser-60351.patch ajouté

- Fichier 0001-api-use-StringOrListField-for-recurrence_days-serial.patch ajouté

- Statut changé de En cours à Solution proposée

Quand wcs envoie la chaine '0, 3, 5', elle est reçu telle qu'elle par le serializer, mais ce n'est pas le cas avec une simple requête request (sans Content-type': application/json dans les headers comme le fait wcs) ou curl.
Je garde donc ma rustine, mais en la précisant pour qu'elle n'attrape que les tableaux contenant une unique chaîne de caractères.

0001: remplacer par StringOrListField et ensuite réparer
0002: ne plus planter sur la trace ci-dessus (et accepter l'entier).

#10 - 03 février 2022 08:48 - Nicolas Roche

- Fichier 0002-api-accept-int-on-StringOrListField-serialiser-60351.patch ajouté

- Fichier 0001-api-use-StringOrListField-for-recurrence_days-serial.patch ajouté

J'ai complété le test avec la valeur retournée.

#11 - 07 février 2022 16:10 - Valentin Deniaud

Nicolas Roche a écrit :

> Quand wcs envoie la chaine '0, 3, 5', elle est reçu telle qu'elle par le serializer, mais ce n'est pas le cas avec une simple requête request (sans Content-type': application/json dans les headers comme le fait wcs) ou curl.

> Je garde donc ma rustine, mais en la précisant pour qu'elle n'attrape que les tableaux contenant une unique chaîne de caractères.

Je pense qu'il ne faut pas jouer à inventer des problèmes dans les tests, et ne pas garder cette rustine.

On doit valider la liste ou la chaîne de caractère qui nous est passée ; on ne doit pas accepter de liste de liste cheloue.

> pour éviter la régression sur ce test : tests/api/test_event.py::test_add_event_serialiser

On peut considérer qu'ici c'est le test qui est faux, et qu'il faut utiliser app.post_json ce qui devrait lui permettre de passer.

Désolé en tout cas d'avoir immédiatement embrouillé le ticket en demandant un changement de serializer, tu as toujours la possibilité de ragequit en tapant genre

```
@@ -32,2 +32,4 @@ class CommaSeparatedStringField(serializers.ListField):
     def to_internal_value(self, data):
+        if not isinstance(data, str):
+            raise ValidationError('input must be a string')
         data = [s.strip() for s in data.split(',') if s.strip()]
```

ce qui aurait été le bon fix simple si je n'avais rien dit.

#12 - 14 février 2022 17:38 - Nicolas Roche

- Fichier 0002-api-use-StringOrListField-for-recurrence_days-serial.patch ajouté

- Fichier 0001-tests-change-post-patch-to-pass-json-to-event-view-6.patch ajouté

> On peut considérer qu'ici c'est le test qui est faux, et qu'il faut utiliser app.post_json ce qui devrait lui permettre de passer.

En effet, merci pour cette indication.

#13 - 15 février 2022 09:42 - Valentin Deniaud

- Statut changé de Solution proposée à Solution validée

Nickel

#14 - 15 février 2022 12:05 - Nicolas Roche

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

```
commit ac41d09c94716e154cacaad5c5fb7207a39158d1
Author: Nicolas ROCHE <nroche@entrouvert.com>
Date:   Wed Feb 2 18:53:31 2022 +0100

    api: use StringOrListField for recurrence_days serialiser (#60351)

commit e532eefc0e11ef9ab48965b87cd81c5610cf6891
Author: Nicolas ROCHE <nroche@entrouvert.com>
Date:   Mon Feb 14 17:19:03 2022 +0100

    tests: change post/patch to pass json to event view (#60351)
```

#15 - 17 février 2022 20:17 - Transition automatique

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

#16 - 24 avril 2022 04:42 - Transition automatique

Automatic expiration

Fichiers

0002-api-improve-recurrence_days-serialiser-on-create-eve.patch	3,1 ko	27 janvier 2022	Nicolas Roche
0001-test-add-a-recurent-event-providing-int-as-recurrent.patch	2,54 ko	27 janvier 2022	Nicolas Roche
0002-api-accept-int-on-recurrence_days-serialiser-60351.patch	2,63 ko	02 février 2022	Nicolas Roche

0001-api-use-StringOrListField-for-recurrence_days-serial.patch	3,35 ko	02 février 2022	Nicolas Roche
0002-api-accept-int-on-StringOrListField-serialiser-60351.patch	2,58 ko	03 février 2022	Nicolas Roche
0001-api-use-StringOrListField-for-recurrence_days-serial.patch	3,56 ko	03 février 2022	Nicolas Roche
0001-tests-change-post-patch-to-pass-json-to-event-view-6.patch	11,2 ko	14 février 2022	Nicolas Roche
0002-api-use-StringOrListField-for-recurrence_days-serial.patch	3,04 ko	14 février 2022	Nicolas Roche