

Corbo - Development #10794

api d'abonnement et desabonnement aux categories par mail

02 mai 2016 11:42 - Serghei Mihai

Statut:	Fermé	Début:	02 mai 2016
Priorité:	Haut	Echéance:	
Assigné à:	Serghei Mihai	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	
Patch proposed:	Oui		
Description			
Conforme au protocole défini par la cellule "newsletter" de combo			

Révisions associées

Révision e93ea142 - 12 juillet 2016 10:06 - Serghei Mihai

api: newsletters retrieval endpoint (#10794)

Révision 8759c981 - 12 juillet 2016 10:06 - Serghei Mihai

api: subscriptions management endpoint (#10794)

Historique

#1 - 03 mai 2016 10:17 - Serghei Mihai

- Fichier 0001-api-newsletters-retrieval-endpoint-10794.patch ajouté
- Fichier 0002-api-subscriptions-management-endpoint-10794.patch ajouté
- Statut changé de Nouveau à En cours
- Patch proposed changé de Non à Oui

Avec les tests lancés par tox pour django-1.7 et 1.8

#2 - 03 mai 2016 11:18 - Benjamin Dauvergne

Déjà j'utiliserai django-rest-framework sauf si tu veux redévelopper du code de validation de signature.

Pour le reste on colle trop à l'existant, donc je ne pense pas que la notion de canal soit utile, on devrait l'avoir déjà dans identifier en utilisant des schémas d'uri (tel:, mail:, etc..), et on se retrouve avec SubscriptionType et Subscription qui font la même chose. Mais c'est le futur, pour l'instant on va ignorer complètement les SMS on verra plus tard.

Il faut un seul modèle Subscription:

```
class Subscription(models.Model):
    category = FK(Category)
    identifiant = URLField(blank=True)
    uuid = CharField(max_length=128, blank=True)
```

dans les appels au web-service si uuid est passé on l'enregistre, sinon on ne met rien. Lors des notifications de provisionning on pourra chercher des abonnements par uuid et éventuellement corriger les abonnements de type mailto: ou tel: en fonction. Pour l'abonnement homepage (en fait la génération d'un flux RSS) on met uuid et pas d'identifiant.

Donc j'étendrai la cellule abonnement de combo pour envoyer aussi le uuid dans la query string, ensuite dans get_subscriptions j'utiliserai le queryset Subscription.objects.filter(Q(identifiant="mailto:" + email)|Q(uuid=uuid)) dans tous les cas je mets homepage dans les transports (si on est abonné il n'y a aucune raison de ne pas voir les news sur sa homepage) et si subscription.identifiant == "mailto:" + email je renvoi aussi le transport "email". Attention à bien utiliser get_or_create(), s'il existe déjà un abonnement pour cet email mais sans uuid, on le met à jour avec l'uuid.

Dans update_subscription idem, si j'ai email je crée un abonnement avec un identifiant mailto: et un uuid (si j'en ai un) et si j'ai juste homepage je ne crée un abonnement qu'avec uuid, si aucun transport je supprime toutes les abonnements ayant cet email ou cet uuid pour la catégorie concernée.

#3 - 04 mai 2016 15:16 - Serghei Mihai

- Fichier 0001-api-newsletters-retrieval-endpoint-10794.patch ajouté
- Fichier 0002-api-subscriptions-management-endpoint-10794.patch ajouté

Ok

#4 - 04 mai 2016 17:15 - Benjamin Dauvergne

mobile et mail ne sont pas des identifiants de schéma d'URL valides, j'enlèverai complètement le fichier channels.py, j'ignorerai complètement qu'on pourrait un jour envoyer des SMS et je mettrai en dur les deux transports qui existent vraiment, soit "homepage" et "email".

```
class NewslettersView(APIView):
    def get(self, request):
        newsletters = []
        transports = [{'id': 'email', 'text': _('Email')}, {'id': 'homepage', 'text': _('Homepage')}]
        for c in Category.objects.all():
            newsletter = {'id': str(c.pk), 'text': c.name, 'transports': transports}
            newsletters.append(newsletter)
        return Response({'data': newsletters})
```

Si email alors identifier = "mailto:" + email (on met ce code en dur dans update_subscriptions), voilà tout.

```
def update_subscriptions(self, category_id, transports, email=None, uuid=None):
    uuid = uuid or u''
    identifier = u'mailto:' + (email or '')
    try:
        cat = Category.objects.get(pk=category_id)
    except Category.DoesNotExist:
        return

    subscriptions = cat.subscription_set

    if email:
        if 'email' in transports:
            subscriptions.get_or_create(identifier=identifier, uuid=uuid)
            if uuid: # delete simple other email subscriptions
                subscriptions.exclude(uuid=uuid).filter(identifier=identifier).delete()
        else:
            subscriptions.filter(identifier=identifier).delete()
    if uuid:
        if 'homepage' in transports:
            subscriptions.get_or_create(uuid=uuid)
        else: # delete simple homepage subscriptions
            subscriptions.filter(identifier='', uuid=uuid).delete()
```

Il faudrait un index unique sur (category, identifier, uuid) sinon les get_or_create() vont foirer. Il faudrait mettre le décorateur @atomic sur cette méthode.

#5 - 04 mai 2016 18:20 - Serghei Mihai

Benjamin Dauvergne a écrit :

mobile et mail ne sont pas des identifiants de schéma d'URL valides, j'enlèverai complètement le fichier channels.py, j'ignorerai complètement qu'on pourrait un jour envoyer des SMS et je mettrai en dur les deux transports qui existent vraiment, soit "homepage" et "email".

J'imaginai les classes dans channels.py avec ses méthodes send qui implémenterait chacune son mode d'envoi, mais ok, partons sans pour l'instant

#6 - 04 mai 2016 18:26 - Benjamin Dauvergne

Non mort à l'abstraction.

#7 - 04 mai 2016 19:08 - Serghei Mihai

- Fichier 0001-api-newsletters-retrieval-endpoint-10794.patch ajouté

- Fichier 0002-api-subscriptions-management-endpoint-10794.patch ajouté

Ok

#8 - 04 mai 2016 19:29 - Benjamin Dauvergne

Le champ channel sur Broadcast et AnnounceForm ne m'a plus l'air utile non plus, pour l'instant on broadcast vers des mails et c'est tout.

#9 - 04 mai 2016 19:51 - Serghei Mihai

- Fichier 0002-api-subscriptions-management-endpoint-10794.patch ajouté

Yep, je le vire dans le 0002

#10 - 04 mai 2016 19:58 - Serghei Mihai

- Fichier 0002-api-subscriptions-management-endpoint-10794.patch supprimé

#11 - 04 mai 2016 19:58 - Serghei Mihai

- Fichier 0002-api-subscriptions-management-endpoint-10794.patch ajouté

le bon patch

#12 - 06 juin 2016 22:49 - Serghei Mihai

- Statut changé de *En cours* à *Résolu* (à déployer)

```
commit d0250b73c4ac2687e6e7dc27baeb261ce2f76559
Author: Serghei Mihai <smihai@entrouvert.com>
Date: Mon May 2 15:32:52 2016 +0200
```

```
api: subscriptions management endpoint (#10794)
```

#13 - 22 février 2017 18:33 - Serghei Mihai

- Statut changé de *Résolu* (à déployer) à *Fermé*

Fichiers

0001-api-newsletters-retrieval-endpoint-10794.patch	20 ko	03 mai 2016	Serghei Mihai
0002-api-subscriptions-management-endpoint-10794.patch	10,7 ko	03 mai 2016	Serghei Mihai
0001-api-newsletters-retrieval-endpoint-10794.patch	9,15 ko	04 mai 2016	Serghei Mihai
0002-api-subscriptions-management-endpoint-10794.patch	11,2 ko	04 mai 2016	Serghei Mihai
0001-api-newsletters-retrieval-endpoint-10794.patch	11 ko	04 mai 2016	Serghei Mihai
0002-api-subscriptions-management-endpoint-10794.patch	10,8 ko	04 mai 2016	Serghei Mihai
0002-api-subscriptions-management-endpoint-10794.patch	13,8 ko	04 mai 2016	Serghei Mihai