

Combo - Development #7994

cellules "items actifs" et "items passés"

03 août 2015 10:10 - Serghei Mihai (congé, retour 15/05)

Statut:	Fermé	Début:	03 août 2015
Priorité:	Haut	Echéance:	
Assigné à:	Serghei Mihai (congé, retour 15/05)	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	
Patch proposé:	Oui		
Description			

Historique

#1 - 04 août 2015 14:59 - Serghei Mihai (congé, retour 15/05)

- Fichier 0001-contrib-teamnet-family-invoices-cell-7994.patch ajouté

- Statut changé de Nouveau à En cours

- Patch proposé changé de Non à Oui

Depend du [#7867](#)

#2 - 14 août 2015 11:09 - Serghei Mihai (congé, retour 15/05)

Après discussion avec Thomas, on préfère que ça soit lingo qui gère cette partie.

Un draft dans <http://git.entrouvert.org/lingo.git?h=teamnet>

#3 - 14 août 2015 11:30 - Frédéric Péters

Il faut alors changer le projet dans ce ticket et, comme il s'agit d'un unique patch, il peut être attaché au ticket.

#4 - 14 août 2015 12:39 - Frédéric Péters

- Projet changé de Combo à Lingo

#5 - 14 août 2015 13:11 - Frédéric Péters

D'une première lecture.

Les deux migrations devraient sans doute être réunies.

```
+ ws_url = models.URLField(_('Webservice URL'), blank=True)
```

ws_url ? faudrait sans doute être plus spécifique, webservice pour faire quoi ?

```
+ def get_invoices(self, nameid):
+     url = self.ws_url + '/regie/%s/invoices/?orig=combo&NameID=%s&months=12' % (self.slug,
+                                                                                   nameid)
```

Notre objet "régie", il se trouve là maintenant dépendant d'un "ws_url" pour connaître la liste des factures; j'ai l'impression qu'il manque une abstraction quelque part. Ça rejoint sans doute la question précédente; de la même manière qu'on a une abstraction pour le service de paiement, on en aurait une ici pour le "service de facturation" (?). (et donc → invoice_webservice_url ?).

```
+ url = self.ws_url + '/regie/%s/invoices/?orig=combo&NameID=%s&months=12' % (self.slug,
+                                                                                   nameid)
+ url = sign_url(url, key='xxx')
```

Si on met des valeurs bidons pour accélérer le développement initial (orig=combo, key=xxx), il faut au moins penser à mettre un gros TODO pour corriger ça plus tard.

```
+ def get_invoices(self, nameid):
+     url = self.ws_url + '/regie/%s/invoices/?orig=combo&NameID=%s&months=12' % (self.slug,
+                                                                                   nameid)
```

Le "months=12", j'en ferais un paramètre.

```

+         url = self.ws_url + ??? + '/?orig=combo&NameID=%s' % (nameid)
+         url = sign_url(url, key='xxx')
+         return requests.get(url).json()
...
+         url = self.ws_url + ??? + '/?orig=combo&NameID=%s' % (nameid)
+         url = sign_url(url, key='xxx')
+         return requests.get(url).json()
...
+         url = self.ws_url + ??? + '/?orig=combo&NameID=%s' % (nameid)
+         url = sign_url(url, key='xxx')
+         return requests.get(url).json()

```

Il me saute aux yeux une factorisation possible.

```

-         return Regie.objects.count() > 0
+         return Regie.objects.filter(ws_url='').count() > 0

```

Ça me semble très très bizarre, ça lie la gestion du panier à l'absence de webservice de facturation. S'il s'agit de faire un truc tout à fait différent, sans rien en commun, autant le faire ailleurs plutôt que mélanger du code sans rapport.

```
+class LingoRemoteRegie(CellBase):
```

Ça me semble plutôt être une cellule "Factures" qu'une cellule "régie distante" (ce qui parle à personne).

```

+         if request.GET.get('number'):
+             context.update({'invoice': self.regie.get_invoice(request.GET['number'],
+                                                             nameid)})
+         else:
+             context.update({'invoices': self.regie.get_invoices(nameid)})

```

Ça sonne comme si la même cellule est à la fois la cellule affichant une liste de factures et la cellule affichant une seule facture, j'évitais ce mélange, qu'il y ait une cellule affichant la liste des factures, point.

```
+         <a href="{ page.get_online_url }?number={ invoice.nofacture }" class="view">{% trans "View" %}</a>
```

En lien avec la note précédente, ce que je ferais ici, c'est ouvrir dans une popup le détail de la facture, plutôt qu'altérer le contenu de la cellule.

```

+<h2>{% trans "Your invoices for" %} {{ regie }}</h2>
...
+<h2>{% trans "Invoice nr. " %} {{ invoice.nofacture }}</h2>

```

On ne peut pas traduire les choses ainsi, dans certaines langues le "regie" ou le "numéro de facture" ne se trouveraient pas à la fin.

```

+         <td>{{ invoice.nofacture }}</td>
+         <td>{{ invoice.femission }}</td>
+         <td>{{ invoice.fmontant }} €</td>

```

Ce serait mieux d'avoir des attributs correspondant à l'anglais des libellés de colonne (no, date, amount) (et label et balance).

```

+         <td>{{ invoice.fmontant }} €</td>
...
+<p><strong>{% trans "Amount:" %}</strong> {{ invoice.fmontant }}€</p>

```

D'un côté il y a un espace entre le montant et l'€, pas dans l'autre.

```

-from .views import RegiesAPIView, AddBasketItemAPIView, PayView, CallbackView
+from .views import RegiesAPIView, AddBasketItemAPIView, PayView, CallbackView, \
+     InvoiceDownloadView

```

Utiliser des parenthèses plutôt que le \.

```

+class InvoiceDownloadView(View):
+     http_method_names = ['get']
+
+     def get(self, request, *args, **kwargs):
+         mellon = self.request.session.get('mellon_session')
+         if mellon:

```

Ici, je ne suis fan de l'arrivée du mellon_session, et là aussi je vivrais très bien la présence d'une couche intermédiaire.

On pourrait avoir "regie.download_invoice(context, kwargs['invoice_id'])", et que ça soit au download_invoice de faire sa popote pour retrouver le nécessaire selon l'utilisateur connecté.

```
+         invoice = regie.download_invoice(kwargs['invoice_id'], mellon['name_id_content'])
```

Et c'est le download_invoice qui, selon moi, doit r aler et lever une exception si la facture demand ee ne correspond pas   l'utilisateur.

Et faire le d ecodage base64.

#6 - 09 septembre 2015 10:33 - Serghei Mihai (cong es, retour 15/05)

- Fichier 0001-active-and-history-items-cells.patch ajout 

- Sujet chang  de cellule "factures" pour teamnet/axel   cellules "factures actives" et "historique des factures"

2 cellules s epar es pour les factures   regler et pour les factures pass es.

#7 - 09 septembre 2015 11:09 - Serghei Mihai (cong es, retour 15/05)

- Sujet chang  de cellules "factures actives" et "historique des factures"   cellules "items actifs" et "items pass s"

#8 - 21 septembre 2015 08:12 - Serghei Mihai (cong es, retour 15/05)

- Fichier 0001-active-and-history-items-cells.patch supprim 

#9 - 21 septembre 2015 08:13 - Serghei Mihai (cong es, retour 15/05)

- Fichier 0001-active-and-history-items-cells-7994.patch ajout 

#10 - 21 septembre 2015 08:13 - Serghei Mihai (cong es, retour 15/05)

- Fichier 0001-contrib-teamnet-family-invoices-cell-7994.patch supprim 

#11 - 22 septembre 2015 09:58 - Serghei Mihai (cong es, retour 15/05)

- Statut chang  de En cours   R esolu (  d eployer)

```
commit 764199ed2aade2a7a82c882523c0c8ca2e84d94d
Author: Serghei Mihai <smihai@entrouvert.com>
Date: Fri Sep 4 11:39:20 2015 +0200
```

```
active and history items cells (#7994)
```

#12 - 08 f evrier 2016 11:31 - Serghei Mihai (cong es, retour 15/05)

- Statut chang  de R esolu (  d eployer)   Ferm 

Fichiers

0001-active-and-history-items-cells-7994.patch

11,1 ko 21 septembre 2015

Serghei Mihai (cong es, retour 15/05)