

Combo - Development #32967

Commencer les options de transactions sur les régies

10 mai 2019 10:57 - Emmanuel Cazenave

Statut:	Fermé	Début:	10 mai 2019
Priorité:	Normal	Echéance:	
Assigné à:	Emmanuel Cazenave	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
Suite de #32441 , sur une régie dont le backend de paiement le supporte, offrir la possibilité configurer la régie pour que les transaction qu'elles génèrent aient le mode de validation manuel.			
Demandes liées:			
Lié à Combo - Development #32441: Nouvel objet PaymentBackend		Fermé	18 avril 2019
Lié à EOPayment - Development #32969: Surcharger dynamiquement des parametres...		Fermé	10 mai 2019

Révisions associées

Révision 23f13059 - 16 mai 2019 17:53 - Emmanuel Cazenave

lingo: start transaction options (#32967)

Based on the backend capabilities.

Historique

#1 - 10 mai 2019 10:57 - Emmanuel Cazenave

- Lié à Development #32441: Nouvel objet PaymentBackend ajouté

#2 - 10 mai 2019 11:13 - Emmanuel Cazenave

- Lié à Development #32969: Surcharger dynamiquement des parametres d'une transaction ajouté

#3 - 10 mai 2019 15:17 - Emmanuel Cazenave

- Fichier 0001-wip.patch ajouté
- Statut changé de Nouveau à En cours
- Patch proposed changé de Non à Oui

Histoire de donner un avant goût.

Ça stocke les options d'une régie sur un champ json transaction_option que l'utilisateur ne voit pas, à la place génération dynamique de champs de formulaire en se basant sur ce les backend proposent ([#32969](#))

#4 - 10 mai 2019 20:31 - Emmanuel Cazenave

- Fichier 0001-lingo-start-transaction-options-32967.patch ajouté
- Sujet changé de Regie de paiement à validation manuelle à Commencer les options de transactions sur les régies
- Statut changé de En cours à Solution proposée

Avec des tests.

#5 - 14 mai 2019 16:44 - Emmanuel Cazenave

- Fichier 0001-lingo-start-transaction-options-32967.patch ajouté

Mis à jour par rapport aux changements dans [#32969](#).

C'est modeste, pas d'ajax et donc un temps de retard en cas de changement de backend, les options permises par le nouveau backend n'apparaissent pas tout de suite il sauve une première fois puis éditer à nouveau ... (je vais voir gérer l'ajax en espérant que ça me prenne pas trois

plombes).

#6 - 15 mai 2019 19:37 - Emmanuel Cazenave

- Fichier 0001-lingo-start-transaction-options-32967.patch ajouté

Option de ne pas présenter ces champs dynamiques lors de la création d'une régie (ainsi que les autres champs standards peu utilisés), tous les champs présents lors de l'édition.

En se disant que changer le backend de paiement d'une régie existante est un cas marginal, et qu'on s'autorise donc à éviter ajax and co.

Également terrain mieux préparé si on veut se mettre à faire la même chose sur les payment backend.

#7 - 16 mai 2019 16:18 - Frédéric Péters

(ainsi que les autres champs standards peu utilisés)

J'en serais à également retirer le `is_default`, je regarde le `save()` de la Regie et ça me semble pouvoir marcher tout seul.

```
combo/apps/lingo/forms.py
```

Y manque le préambule de licence.

```
if param['type'] is bool:
```

On mettrait un `else: raise NotImplementedError` pour être sûr ?

```
super(RegieForm, self).save()
```

Je préfère la forme `instance = super(RegieForm, self).save()`, puis agir sur `instance` (plutôt que `self.instance`) plutôt que dépendre du fait qu'en interne l'attribut s'appelle "instance".

#8 - 16 mai 2019 16:23 - Frédéric Péters

~~Je préfère la forme `instance = super(RegieForm, self).save()`, puis agir sur `instance` (plutôt que `self.instance`) plutôt que dépendre du fait qu'en interne l'attribut s'appelle "instance".~~

Sauf que je me dis que ça ne tient pas la route, que je parlais de l'idée que ça retournerait l'objet alors que ce n'est peut-être bien pas le cas.

#9 - 16 mai 2019 18:03 - Emmanuel Cazenave

- Fichier 0001-lingo-start-transaction-options-32967.patch ajouté

Tout pris en compte.

Frédéric Péters a écrit :

J'en serais à également retirer le `is_default`, je regarde le `save()` de la Regie et ça me semble pouvoir marcher tout seul.

Oui ça marche.

Je préfère la forme `instance = super(RegieForm, self).save()`, puis agir sur `instance` (plutôt que `self.instance`) plutôt que dépendre du fait qu'en interne l'attribut s'appelle "instance".

Ça passe aussi.

#10 - 16 mai 2019 18:08 - Frédéric Péters

- Statut changé de Solution proposée à Solution validée

#11 - 16 mai 2019 18:10 - Emmanuel Cazenave

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

commit 23f13059d00a761af113a09b92d0031f9648cddb
Author: Emmanuel Cazenave <ecazenave@entrouvert.com>
Date: Tue May 14 16:38:32 2019 +0200

lingo: start transaction options (#32967)

Based on the backend capabilities.

#12 - 16 mai 2019 19:15 - Frédéric Péters

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

Fichiers

0001-wip.patch	5,52 ko	10 mai 2019	Emmanuel Cazenave
0001-lingo-start-transaction-options-32967.patch	9,31 ko	10 mai 2019	Emmanuel Cazenave
0001-lingo-start-transaction-options-32967.patch	9,04 ko	14 mai 2019	Emmanuel Cazenave
0001-lingo-start-transaction-options-32967.patch	10,3 ko	15 mai 2019	Emmanuel Cazenave
0001-lingo-start-transaction-options-32967.patch	11,9 ko	16 mai 2019	Emmanuel Cazenave