

Combo - Development #8167

possibilité d'ouvrir des popups dans les pages

03 septembre 2015 17:06 - Serghei Mihai (congé, retour 15/05)

Statut:	Fermé	Début:	03 septembre 2015
Priorité:	Normal	Echéance:	
Assigné à:	Serghei Mihai (congé, retour 15/05)	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	
Patch proposé:	Oui		
Description			
De la même façon que dans /manage, on devrait pouvoir ouvrir des dialogs dans les pages à coup de <code>lien</code>			
Demandes liées:			
Lié à Combo - Development #8293: Ajouter MediaDefiningClass à la métaclasse d...		Fermé	18 septembre 2015

Révisions associées

Révision 4e94b628 - 21 septembre 2015 11:20 - Serghei Mihai (congé, retour 15/05)

expose cells media in the page (#8167)

Historique

#1 - 09 septembre 2015 12:14 - Serghei Mihai (congé, retour 15/05)

- Fichier 0001-include-gadjo.js-and-its-dependencies-in-pages-8167.patch ajouté
- Statut changé de Nouveau à En cours
- Assigné à mis à Serghei Mihai (congé, retour 15/05)
- Patch proposé changé de Non à Oui

En incluant gadjo.js, jquery.ui dans le theme publik

#2 - 10 septembre 2015 09:30 - Frédéric Péters

Ça m'ennuie un petit peu d'ajouter le chargement de jqueryui pour l'éventualité d'une cellule demandant une popup, idéalement je préférerais un système de dépendances, que la cellule "factures" déclare qu'elle a besoin de gadjo.js et que celui-ci soit alors ajouté.

Un autre aspect est qu'avec la modification ici on va charger gadjo.js systématiquement et il va se trouver chargé aussi dans le thème partagé avec wcs, qui a son propre handler pour les `a[rel=popup]`, c'est quelque chose qu'on veut éviter.

Un détail pour finir, il faudrait modifier gadjo.js pour que le code gérant le panneau latéral ne soit pas exécuté si les éléments ne sont pas présents (histoire de ne pas ajouter des choses à window.localStorage).

#3 - 10 septembre 2015 18:04 - Serghei Mihai (congé, retour 15/05)

- Fichier 0001-allow-cells-declare-pop-ups-use-8167.patch ajouté

Les cellules pourraient déclarer qu'elles ouvrent des pop-ups et dans ce cas, le template de la page chargerait gadjo.js et le reste.

il faudrait modifier gadjo.js pour que le code gérant le panneau latéral ne soit pas exécuté si les éléments ne sont pas présents

je fais un autre ticket

#4 - 10 septembre 2015 18:09 - Frédéric Péters

J'imaginai un truc un peu plus générique, genre `extra_css_includes`, `extra_js_scripts` (un peu à l'image des `add_javascript()` et `add_css_include()` dans wcs).

#5 - 14 septembre 2015 17:40 - Serghei Mihai (congé, retour 15/05)

- Fichier 0001-allow-cells-add-extra-js-and-css-8167.patch ajouté

Un peu de cette façon, j'imagine?

Ça m'embete un peu de devoir déclarer dans les cellules nécessitant le popup les scripts jquery et la css:

```
extra_css = ['xstatic/themes/smoothness/jquery-ui.min.css']
...
extra_js = ['xstatic/jquery-ui.min.js', 'xstatic/jquery.min.js', 'js/gadjo.js']
```

#6 - 14 septembre 2015 17:44 - Frédéric Péters

Alors tu peux aussi imaginer un système identique au renderblock d'authentic, si tu préfères déclarer ces dépendances dans un template et non dans un fichier python.

#7 - 18 septembre 2015 10:17 - Benjamin Dauvergne

À noter qu'il y a déjà tout ce qu'il faut pour faire cela dans `django.forms.widgets.Media`, ce n'est pas lié à l'implémentation de Form et parfaitement réutilisable. Il faudrait aussi importer l'implémentation de `media_property` pour un champ media et la metaclass `MediaDefiningClass`.

Ça devrait permettre un truc du genre:

```
ctx['media'] = sum(cell.media for cell in cells)
```

dans la vue des pages.

#8 - 18 septembre 2015 10:18 - Benjamin Dauvergne

Ensuite plutôt que de déclarer dans chaque cellule faisant des popups ces scripts on peut faire une classe de base `CellWithPopup` qui le déclare.

#9 - 18 septembre 2015 10:40 - Benjamin Dauvergne

- Lié à [Development #8293](#): Ajouter `MediaDefiningClass` à la metaclass de `CellBase` ajouté

#10 - 18 septembre 2015 11:33 - Serghei Mihai (congé, retour 15/05)

- Fichier `0001-allow-cells-to-declare-medias-8167.patch` ajouté

Suite au [#8293](#)

#11 - 18 septembre 2015 11:37 - Serghei Mihai (congé, retour 15/05)

- Fichier `0001-cell-allowing-pop-ups-opening-8167.patch` ajouté

Et comme le suggère Benj, une cellule permettant d'ouvrir des popups

#12 - 18 septembre 2015 13:45 - Frédéric Péters

Je ne suis pas fan, je préfère commencer en gardant les trois lignes dans la classe qui les utilise, et éviter pour le moment le coût de compréhension amené par une classe supplémentaire. (mais je ne vais pas discuter de ça longtemps) (par contre le sujet du commit est à revoir, ce n'est pas une cellule qui est ajoutée mais une classe, je préférerais un truc genre "factor media declaration for popups in a different class").

#13 - 19 septembre 2015 10:31 - Serghei Mihai (congé, retour 15/05)

Pas de problème. On peut très bien se débrouiller en ajoutant le class `Media` dans les classes qui en besoin

#14 - 21 septembre 2015 13:09 - Serghei Mihai (congé, retour 15/05)

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

```
commit 4e94b62894e5cf8fab7094e9c1897008152ecfab
Author: Serghei Mihai <smihai@entrouvert.com>
Date: Thu Sep 10 17:59:25 2015 +0200
```

```
expose cells media in the page (#8167)
```

#15 - 29 novembre 2015 22:21 - Frédéric Péters

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

Fichiers

<code>0001-include-gadjo.js-and-its-dependencies-in-pages-8167.patch</code>	1,56 ko 09 septembre 2015	Serghei Mihai (congé, retour 15/05)
<code>0001-allow-cells-declare-pop-ups-use-8167.patch</code>	2,51 ko 10 septembre 2015	Serghei Mihai (congé, retour 15/05)

0001-allow-cells-add-extra-js-and-css-8167.patch
0001-allow-cells-to-declare-medias-8167.patch
0001-cell-allowing-pop-ups-opening-8167.patch

2,34 ko 14 septembre 2015
1,63 ko 18 septembre 2015
949 octets 18 septembre 2015

Serghei Mihai (congés, retour 15/05)
Serghei Mihai (congés, retour 15/05)
Serghei Mihai (congés, retour 15/05)