

w.c.s. - Development #36627

rendu odt de {{form_details}}

02 octobre 2019 15:55 - Frédéric Péters

Statut:	Fermé	Début:	02 octobre 2019
Priorité:	Normal	Echéance:	16 mars 2020
Assigné à:	Frédéric Péters	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
{{form_details}} dans un modèle de document, tout se trouve posé dans un unique paragraphe, avec des retours à la ligne forcés là où des lignes blanches sont vues. (mécanisme qui avait été prévu pour les commentaires laissés par un usager dans un champ texte).			
Il faudrait plutôt détecter le cas particulier de {{form_details}} (on générerait uniquement le cas où celui-ci est dans un champ avec rien d'autre), et gérer une structure odt correspondante.			
Demandes liées:			
Lié à w.c.s. - Development #31712: Les champs titres et sous-titre devraient ...		Fermé	25 mars 2019
Lié à w.c.s. - Bug #36099: form_details dans un document ODT, n'affiche pas t...		Fermé	13 septembre 2019

Révisions associées

Révision 9bd888c7 - 17 mars 2020 15:03 - Frédéric Péters

factor out function to clean text for open document (#36627)

Révision 0441839f - 17 mars 2020 15:03 - Frédéric Péters

misc: add open document representation of fields (#36627)

Révision c5a08111 - 17 mars 2020 15:03 - Frédéric Péters

workflows: add support for form details odt section in documents (#36627)

Historique

#1 - 02 octobre 2019 15:56 - Frédéric Péters

- Lié à Development #31712: Les champs titres et sous-titre devraient être dans form_details ajouté

#2 - 07 octobre 2019 10:37 - Mikaël Ates (de retour le 29 avril)

- Lié à Bug #36099: form_details dans un document ODT, n'affiche pas tout le formulaire ajouté

#3 - 07 février 2020 17:01 - Frédéric Péters

- Echéance mis à 01 juin 2020

- Assigné à mis à Frédéric Péters

(sauf si ça motive quelqu'un avant)

#4 - 08 février 2020 11:10 - Frédéric Péters

J'ai posé une branche <https://git.entrouvert.org/wcs.git/log/?h=wip/36627-odt-form-details> qui expose le plan, partager le code déterminant les champs à afficher (fait, et sans doute à intégrer ~rapidement dans master parce que sinon ça va être pénible à rebaser), ajout d'une représentation odt aux champs (get_opendocument_node_value, qu'il faudra créer pour les différents types de champs qu'on veut pouvoir représenter), puis exploitation de ça pour taper un "form_details" sous forme de section dans l'odt (libreoffice, insérer, section, donner le nom form_details), partie qu'il faudra compléter d'un peu de style (espacement entre les champs etc.).

#5 - 16 février 2020 17:11 - Frédéric Péters

- Echéance changé de 01 juin 2020 à 16 mars 2020

#6 - 03 mars 2020 14:23 - Frédéric Péters

- Fichier 0003-misc-add-open-document-representation-of-fields-3662.patch ajouté
- Fichier 0002-factor-out-function-to-clean-text-for-open-document-.patch ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui
- Fichier 0004-workflows-add-support-for-form-details-odt-section-i.patch ajouté

Ça vient après [#39739](#) que j'avais isolé pour lui permettre une relecture plus rapide.

#7 - 03 mars 2020 14:40 - Frédéric Péters

- Fichier template-form-details.odt ajouté
- Fichier Capture d'écran de 2020-03-03 14-36-46.png ajouté

Pour la documentation :

- passer par insertion / section, lui donner form_details comme nom (cf capture)
 - si du contenu est ajouté à la section, il sera retiré.
- pour les styles, ils sont Page_Title (titre de page), Form_Title (champ titre), Form_Subtitle (champ sous-titre), Field_Label (libellé de champ) et Field_Value (valeur de champ)
- si aucun style n'est défini, des styles basiques par défaut sont utilisés
 - dans l'exemple en pièce jointe, les différents styles sont définis, via un exemplaire dans la section, il peut être nécessaire de procéder ainsi (avoir du texte utilisant les styles dans la section) pour éviter un nettoyage automatique des styles inutilisés.

#8 - 17 mars 2020 13:52 - Thomas Noël

Sans trop de difficulté je pense que tu pourrais ajouter dans le test de rendu une page vide, et vérifier qu'elle n'est pas là. Quelque chose comme :

```

+         TableField(id='12', label='Table', type='table', columns=['a', 'b'], rows=['c', 'd']),
+         PageField(id='13', label='Invisible Page', type='page'),
+         TitleField(id='14', label='Invisible Title', type='title'),
+         StringField(id='15', label='Invisible String', type='string', varname='invisiblestr'),
+
+ ...
+
+         assert new_content.count('/table:table-cell') == 8
+         assert 'Invisible' not in new_content

```

J'ai presque envie de proposer qu'au lieu de « node.attrib.get(SECTION_NAME) 'form_details' » on teste carrément « node.attrib.get(SECTION_NAME).replace(';',) '{{form_details}}' » ... sans se donner la peine d'expliquer que ça ne marche que pour cette seule variable, juste parce que j'imagine déjà que beaucoup de monde va taper {{form_details}} dans le nom de la section et demander de l'aide parce que ça marchera pas.

Sinon rien d'autre à ajouter, tout me semble bon. Juste dommage que LibreOffice ne propose pas un mode où le nom de la section s'affiche dans un coin (mais ça n'a pas de rapport avec ce patch).

#9 - 17 mars 2020 14:39 - Frédéric Péters

Branche à jour avec ces changements poussée.

#10 - 17 mars 2020 14:52 - Thomas Noël

- Statut changé de Solution proposée à Solution validée

#11 - 17 mars 2020 15:04 - Frédéric Péters

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

```

commit c5a081113153b4b1b399b3efe85a81f62ea81700
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Sat Feb 8 11:01:17 2020 +0100

```

```

workflows: add support for form details odt section in documents (#36627)

```

```

commit 0441839f27ed769d382e486c5bf757d5b1daccf9
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Sat Feb 8 10:31:14 2020 +0100

```

```

misc: add open document representation of fields (#36627)

```

```

commit 9bd888c708c8258bf2bf92d07f52b53e768dd5b3
Author: Frédéric Péters <fpeters@entrouvert.com>

```

Date: Sat Feb 8 10:30:39 2020 +0100

factor out function to clean text for open document (#36627)

#12 - 17 mars 2020 18:15 - Frédéric Péters

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

#13 - 19 mars 2020 06:16 - Pierre Cros

Je teste, j'arrive à jouer avec les champs titre et sous-titre, en revanche j'ai rien dans Page_Title. D'où vient cette valeur ?

Autre question, peut-être déjà posée, je pense qu'il n'est pas possible d'avoir un truc pour les champs du genre :

Libellé : valeur du champ

#14 - 19 mars 2020 08:18 - Frédéric Péters

(...) en revanche j'ai rien dans Page_Title. D'où vient cette valeur ?

Tu veux dire que les noms de page ne sont pas repris ?

Autre question, peut-être déjà posée, je pense qu'il n'est pas possible d'avoir un truc pour les champs du genre :

Libellé : valeur du champ

Non; c'est déjà bien compliqué d'obtenir un peu de contrôle sur le style des paragraphes, gérer du style à l'intérieur de ceux-ci pas moyen.

#15 - 19 mars 2020 08:52 - Pierre Cros

J'avais juste pas compris que Page_Title = Libellé d'un champ de type page (je testais avec un formulaire sans champ de type Page). Merci.

Compris pour le reste.

Fichiers

0003-misc-add-open-document-representation-of-fields-3662.patch	7,38 ko	03 mars 2020	Frédéric Péters
0004-workflows-add-support-for-form-details-odt-section-i.patch	33,7 ko	03 mars 2020	Frédéric Péters
0002-factor-out-function-to-clean-text-for-open-document-.patch	1,99 ko	03 mars 2020	Frédéric Péters
template-form-details.odt	9,28 ko	03 mars 2020	Frédéric Péters
Capture d'écran de 2020-03-03 14-36-46.png	39,1 ko	03 mars 2020	Frédéric Péters