

w.c.s. - Development #27994

texte riche pour les commentaires (usagers/agents)

14 novembre 2018 14:20 - Frédéric Péters

Statut:	Fermé	Début:	14 novembre 2018
Priorité:	Normal	Echéance:	
Assigné à:	Frédéric Péters	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description via #27990 , Ckeditor devienne aussi l'éditeur pour : Éventuellement l'action de workflow commentaire (mais là il s'agit d'avoir un ckeditor dans l'interface de traitement, y a peut-être des soucis différents. C'est moins important que le reste).			
Demandes liées:			
Lié à w.c.s. - Development #56456: Zone commentaire d'un traitement d'une dem...		Fermé	30 août 2021
Lié à w.c.s. - Development #58808: utiliser bleach pour le nettoyage HTML		Fermé	21 novembre 2021

Révisions associées

Révision 7338e136 - 16 janvier 2022 13:34 - Frédéric Péters

misc: allow "rich" text in comments (#27994)

Révision 0ec6a37b - 16 janvier 2022 14:15 - Frédéric Péters

trivial: make sure an empty form_comment is the empty string (#27994)

Historique

#1 - 03 mai 2019 11:56 - Thomas Noël

À noter qu'il faudra alors gérer l'inclusion de ces commentaires html dans les mails ou documents odt (peut-être à l'aide de filtres à écrire, ou juste |safe pour les mails s'ils sont eux aussi édités en html, cf [#27990](#))

#2 - 03 mai 2019 12:13 - Benjamin Dauvergne

Thomas Noël a écrit :

À noter qu'il faudra alors gérer l'inclusion de ces commentaires html dans les mails ou documents odt (peut-être à l'aide de filtres à écrire, ou juste |safe pour les mails s'ils sont eux aussi édités en html, cf [#27990](#))

Pour ODT c'est juste, comment dire, ... il n'y pas de sous-ensemble d'éléments dans le schéma ODT qui permette une mise en forme simple, tout passe par les styles et donc même avec une lib faite exprès ça donne ça :

https://github.com/lpod/lpod-python-recipes/blob/master/Examples/create_a_text_document_from_plain_text_with_layout.py

```
# master style, using the precedent layout for the actual document
_style_master = odf_create_element(u"""\
<style:master-page style:name="Standard" \
style:page-layout-name="MyLayout"><style:footer><text:p text:style-name="Footer">\
<text:tab/><text:tab/><text:page-number \
text:select-page="current"/> / <text:page-count \
style:num-format="1">15</text:page-count></text:p></style:footer>\
</style:master-page>""")
```

Pas glop.

#3 - 30 août 2021 09:24 - Frédéric Péters

- Lié à Development #56456: Zone commentaire d'un traitement d'une demande // Mise en forme du texte saisi ajouté

#4 - 21 novembre 2021 15:52 - Frédéric Péters

- Lié à Development #58808: utiliser bleach pour le nettoyage HTML ajouté

#5 - 14 décembre 2021 10:44 - Frédéric Péters

- Statut changé de Nouveau à En cours

- Assigné à mis à Frédéric Péters

#6 - 20 décembre 2021 16:01 - Frédéric Péters

Il y a déjà du code qui peut être relu dans la branche, juste ça utilise ckeditor pour tester en attendant #59578.

#8 - 21 décembre 2021 15:24 - Frédéric Péters

- Fichier 0001-misc-allow-rich-text-in-comments-27994.patch ajouté

- Statut changé de En cours à Solution proposée

- Patch proposed changé de Non à Oui

Voilà avec l'intégration godo. Le truc de l'utilisation sous forme de module js, ça fait que l'infra "add_javascript" ne peut pas être utilisée, donc pas de switch entre version minifiée ou pas selon le debug activé ou pas.

À part ça, le truc important c'est que le commentaire est désormais enregistré dans son propre objet (WorkflowCommentPart) plutôt que directement dans un attribut "comment" de l'objet Evolution. Ça correspond à ce qu'on a toujours fait pour les fichiers attachés et les informations ajoutées via l'action "Message dans l'historique" et ce qui est désormais aussi fait pour l'action "formulaire". L'attribut "comment" reste utilisé à la marge ("Administrator reassigned status" et "Form exported in a model") et ça pourra disparaître.

Pour compatibilité avec un éventuel existant le commentaire reste tapé en vrac dans workflow_data, dans une version sans balisage.

Le commentaire était également utilisé de manière automatique dans un {{form_evolution}}, je pense que plus personne n'utilise ça mais pareil une version sans balisage y est posée.

Il y a aussi {{form_comment}} et pareil pas clair que ça soit encore utilisé (chercher form_comment ne donne aucun résultat sur le site de la doc) mais même plan ici, version sans balisage.

#9 - 22 décembre 2021 11:06 - Benjamin Dauvergne

Frédéric Péters a écrit :

Voilà avec l'intégration godo. Le truc de l'utilisation sous forme de module js, ça fait que l'infra "add_javascript" ne peut pas être utilisée, donc pas de switch entre version minifiée ou pas selon le debug activé ou pas.

Je ne comprends pas bien la différence avec un bout de js normal; est-ce que tu vois quelque chose que je peux changer au packaging pour améliorer ça ? Il me semble que leaflet est aussi packagé sous forme xstatic.

#10 - 22 décembre 2021 12:16 - Frédéric Péters

Je ne comprends pas bien la différence avec un bout de js normal.

C'est un "module js". Il ne se trouve pas chargé via un <script src="..."> (gérée via response.add_javascript, qui a du code pour choisir une version minifiée si pas debug); ici il est chargé "inline" par une ligne :

```
import Godo from "/static/xstatic/js/godo.js";
```

(et oui il y aurait moyen d'exposer le booléen comme quoi on est en mode debug et l'utiliser ici pour faire varier le chemin, mon propos est que ça ne matche pas l'infra actuelle) (côté godo j'avais créé [#59585](#) qui couvrait un peu ça) (ça n'est pas bien grave, c'est gzippé 70ko à transférer en plus, ne passons pas de temps ici sur cette partie).

#11 - 14 janvier 2022 10:44 - Benjamin Dauvergne

- Statut changé de Solution proposée à Solution validée

Ok. Je verrai bien l'ajout de strip=True à bleach.clean() à son propre commit, ça n'a pas l'air lié particulièrement à ce nouveau widget ou alors je veux bien une explication (j'ai donné un coup d'œil à la doc¹ de bleach, j'ai l'impression que ça aurait du déjà être la bonne valeur à appliquer avant ce ticket).

¹<https://bleach.readthedocs.io/en/latest/clean.html#bleach.clean>

#12 - 14 janvier 2022 10:48 - Frédéric Péters

Le strip=True est introduit ici parce que c'est ici qu'on introduit une limitation sévère sur les balises autorisées. Comme elles étaient toutes autorisées avant, ça n'avait pas d'incidence.

#13 - 16 janvier 2022 13:34 - Frédéric Péters

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

```
commit 7338e136e5df7ce7a569a479f9cc4731b035c22c
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Tue Dec 14 10:43:44 2021 +0100
```

```
misc: allow "rich" text in comments (#27994)
```

#14 - 16 janvier 2022 16:16 - Frédéric Péters

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

#15 - 20 mars 2022 04:42 - Transition automatique

Automatic expiration

Fichiers

0001-misc-allow-rich-text-in-comments-27994.patch

11,5 ko 21 décembre 2021

Frédéric Péters