

## w.c.s. - Development #23685

### outils de debug intégrés

11 mai 2018 14:42 - Thomas Noël

<b>Statut:</b> Fermé	<b>Début:</b> 11 mai 2018
<b>Priorité:</b> Normal	<b>Echéance:</b>
<b>Assigné à:</b>	<b>% réalisé:</b> 0%
<b>Catégorie:</b>	<b>Temps estimé:</b> 0:00 heure
<b>Version cible:</b>	<b>Planning:</b>
<b>Patch proposed:</b> Oui	
<b>Description</b>	
Sur la page /inspect d'une demande, on aimerait un outil de debug et d'aide au développement qui permette de calculer facilement :	
<ul style="list-style-type: none"><li>• le résultat d'une condition Django</li><li>• le résultat d'une condition Python</li><li>• le résultat d'un gabarit «pour un mail» avec la transformation HTML et texte via docutils</li><li>• le résultat d'un gabarit/expression «sur une ligne»</li><li>• le résultat d'une génération de document OpenOffice</li></ul>	
Idéalement, on peut taper librement des conditions/expressions/gabarit, mais aussi en choisir un parmi ceux disponibles dans le formulaire et le workflow.	
<b>Demandes liées:</b>	
Lié à w.c.s. - Development #30418: ajouter un lien vers l'inspecteur de deman...	<b>Fermé</b> <b>06 février 2019</b>
Lié à w.c.s. - Development #17345: zone de test d'expression sur la page /ins...	<b>Fermé</b> <b>05 juillet 2017</b>
Duplique w.c.s. - Development #14225: dans les ateliers formulaire/workflow :...	<b>Fermé</b> <b>07 décembre 2016</b>

### Révisions associées

#### Révision 27f2c4f2 - 07 février 2019 12:27 - Thomas Noël

add test tools in formdata inspect view (#23685)

### Historique

#### #1 - 23 mai 2018 17:18 - Thomas Noël

- Assigné à mis à Thomas Noël

#### #2 - 31 octobre 2018 12:14 - Brice Mallet

+2 (moi et mon agacement à ne pouvoir me débrouiller seul;-)

#### #3 - 31 octobre 2018 13:07 - Pierre Cros

Thomas, on est tous avec toi :)

#### #4 - 26 novembre 2018 09:24 - Thomas Noël

- Assigné à Thomas Noël supprimé

#### #5 - 06 février 2019 00:28 - Thomas Noël

- Fichier Screenshot\_2019-02-06 Backoffice de Démarches - vide.png ajouté

J'ai commencé un draft sur la branche <https://git.entrouvert.org/wcs.git?h=wip/23685-test-tools-in-inspect-view>

Ca donne ça (copie d'écran). Je suis preneur de remarque sur le code (ça augmente la taille du code de inspect, est-ce qu'on met ça ailleurs, bof) et sur le html parce que c'est pas super joli y'a un gros espace avant le result à cause du bouton submit qu'on voit pas mais qui est là, nécessaire en mode "debug de template".

#### #6 - 06 février 2019 06:13 - Pierre Cros

Ça donne envie.

Perso je connais la page inspect d'une demande (j'imagine que tu parles de celle-là) et la page inspect d'un WF. Mais je ne vois pas comment on

accède à la page inspect tant qu'il n'y a pas de demandes dans le formulaire (qu'on est en train de concevoir et on a alors besoin de l'outil), tu as peut-être déjà pensé à ça. Oui on peut obliger les admins à faire une demande à chaque fois qu'ils veulent tester mais c'est moins bien.

Par ailleurs, laisser ça sur la page d'inspect d'une demande en se disant que ça incitera les administrateurs fonctionnels à aller la voir, ok. Mais alors il faut des liens permanent depuis les fabriques pour aller vers cet outil : depuis la page d'édition des champs côté forms et depuis la page d'accueil d'un statut côté WF.

#### #7 - 06 février 2019 10:00 - Frédéric Péters

tant qu'il n'y a pas de demandes

Oui, l'inspect, et le debug qui arrive ici, s'appliquent sur de vraies données.

Ailleurs, je me mets à imaginer qu'on pourrait créer un faux formdata, avec des données fictives, dans tous les champs (même si ça crée des situations impossibles, genre des champs sur lesquels l'utilisateur ne serait pas passé, et pas les données qui arriveraient via webservice ou en cours de workflow), et qu'on pourrait arriver sur cette page depuis la page inspect d'un formulaire ([#30381](#)) (genre via un lien "inspect d'une demande fictive").

#### #8 - 06 février 2019 10:28 - Thomas Noël

Pierre Cros a écrit :

Perso je connais la page inspect d'une demande (j'imagine que tu parles de celle-là) et la page inspect d'un WF. Mais je ne vois pas comment on accède à la page inspect tant qu'il n'y a pas de demandes dans le formulaire (qu'on est en train de concevoir et on a alors besoin de l'outil), tu as peut-être déjà pensé à ça. Oui on peut obliger les admins à faire une demande à chaque fois qu'ils veulent tester mais c'est moins bien.

Ouais mais pour l'instant il faudra vivre avec ça, difficile de faire autrement. Le plan de Frédéric lié à [#30381](#) est intéressant ceci dit.

Par ailleurs, laisser ça sur la page d'inspect d'une demande en se disant que ça incitera les administrateurs fonctionnels à aller la voir, ok. Mais alors il faut des liens permanent depuis les fabriques pour aller vers cet outil : depuis la page d'édition des champs côté forms et depuis la page d'accueil d'un statut côté WF.

Yep yep c'est le plan, les utilisateurs qui auront accès à une des fabriques (form ou wf) verraient un lien "inspecteur" dans la barre latérale. Je vais faire un ticket, tiens.

#### #9 - 06 février 2019 11:20 - Thomas Noël

- Lié à *Development #30418: ajouter un lien vers l'inspecteur de demande quand on a accès à une des fabriques* ajouté

#### #10 - 06 février 2019 15:44 - Thomas Noël

- Fichier *0001-add-test-tools-in-formdata-inspect-view-23685.patch* ajouté

- Statut changé de *Nouveau* à *Solution proposée*

- Patch proposé changé de *Non* à *Oui*

Voilà, ça marche, c'est pas très très joli mais ça fait (bien) le travail. Il manque un output de "la version mail" d'un gabarit, et la possibilité éventuellement de tester un modèle ODT, ça sera l'objet d'évolutions dans d'autres tickets, plus tard.

#### #11 - 06 février 2019 16:05 - Frédéric Péters

Plutôt des boutons radio qu'un select (et je dis ça après avoir vérifié que c'était bien géré côté js).

Pour la condition c'est possible que ça soit posé sur condition django par défaut ? (ici je n'ai pas vérifié)

Domage d'avoir cette évaluation nécessitant un bouton, à venir rapidement derrière la réécriture pour faire ça à la volée, ou au moins sans demander un rechargement général de la page. Dans cette idée, peut-être déjà taper la création du formulaire dans une méthode particulière, et le `if form.is_submitted():...` dans une autre, qu'il suffira ainsi de déclarer dans le `_q_exports`.

Je trouve aussi dommage de ne pas passer le rendu de cette page en template django mais ok, c'est autre chose.

Côté rendu (et je suis ok pour faire la passe moi-même),

- Plutôt que le titre replié j'aurais bien la zone tout à fait cachée et un bouton "Outils de test" au niveau de la barre de titre, avant "définition du workflow" (et cliquer dessus ferait apparaître la zone)
- Les espacements, les couleurs, etc.

#### #12 - 06 février 2019 16:56 - Thomas Noël

- Fichier *0001-add-test-tools-in-formdata-inspect-view-23685.patch* ajouté

Frédéric Péters a écrit :

Plutôt des boutons radio qu'un select (et je dis ça après avoir vérifié que c'était bien géré côté js).

Plus facile, effectivement.

Pour la condition c'est possible que ça soit posé sur condition django par défaut ? (ici je n'ai pas vérifié)

J'ai pas trouvé et pourtant ça serait vraiment mieux. Presque envie de le faire par défaut prochainement sur le widget... "et voilà"... (mais il faudrait sans doute qu'on mette le petit serpent vert, pour que ça saute bien aux yeux des habitués)

Domage d'avoir cette évaluation nécessitant un bouton, à venir rapidement derrière la réécriture pour faire ça à la volée, ou au moins sans demander un rechargement général de la page. Dans cette idée, peut-être déjà taper la création du formulaire dans une méthode particulière, et le if `form.is_submitted()`:... dans une autre, qu'il suffira ainsi de déclarer dans le `_q_exports`.

Oui alors j'ai pensé très fort à tout ça mais je suis tellement pas doué pour le jajax... donc j'ai déjà fait la séparation comme tu l'indiques, je vois bien comment ça devrait marcher, j'ai pas encore pigé comment faire le post sans recharger la page, ça viendra... Si on peut partir sur cette bêta, ça m'irait bien.

Je trouve aussi dommage de ne pas passer le rendu de cette page en template django mais ok, c'est autre chose.

Ouaip j'y ai pensé quand j'avais déjà écrit 80% du code...

Côté rendu (et je suis ok pour faire la passe moi-même),

- Plutôt que le titre replié j'aurais bien la zone tout à fait cachée et un bouton "Outils de test" au niveau de la barre de titre, avant "définition du workflow" (et cliquer dessus ferait apparaître la zone)

Là j'ai carrément viré le foldable et, si tu veux mon avis, on devrait presque le laisser ainsi, si on peut réduire l'espace. Car la page d'inspect, je la vois bien utilisée principalement pour ça désormais (elle n'était utilisée que par quelques connaisseurs auparavant).

- Les espacements, les couleurs, etc.

Oui, d'accord, humilie moi.

#### #13 - 06 février 2019 18:14 - Frédéric Péters

Je viens d'envoyer dans la branche un commit supplémentaire avec des modifs html/css

<https://git.entrouvert.org/wcs/git/commit/?h=wip/23685-test-tools-in-inspect-view&id=8d2bd5553c80adac4efab1c48b40a6e1862bc5d7>

#### #14 - 06 février 2019 21:35 - Frédéric Péters

En fait passer par un widget condition donne un comportement pas top, avec un bout d'erreur, pour la syntaxe, attrapé et affiché attaché au champ, et d'autres erreurs, affichées dans l'évaluation.

Je viens de mettre un autre commit (dans la branche `wip/23685-test-tools-in-inspect-view-fred`, l'autre avait bougé), qui remplace simplement ça par deux champs texte (condition django et condition python), ça a aussi l'avantage de mettre condition django en premier. Et je sélectionne par défaut condition django.

#### #15 - 06 février 2019 23:18 - Frédéric Péters

+ 1 commit pour avoir le submit ajax.

#### #16 - 07 février 2019 11:26 - Thomas Noël

Super chouette. J'ai repris ça dans ma branche et ajouté un commit qui retire les

«`validation_function=ComputedExpressionWidget.validate_template`», parce que sinon avec le mode ajax on ne voit plus les soucis de validation sur les templates avec erreur de syntaxe. Et c'est plus clair ainsi, toutes les erreurs sont affichées au même endroit, dans les résultats.

#### #17 - 07 février 2019 11:38 - Frédéric Péters

- *Statut changé de Solution proposée à Solution validée*

Ah oui pas trop regardé le côté template mais oui l'idée était bien de pousser toutes les erreurs dans la zone de résultat. (et ensuite bonus ça a facilité l'affaire ajax).

Tous les tests passent, pour moi c'est ok, tu peux squasher les commits et envoyer ça.

**#18 - 07 février 2019 12:28 - Thomas Noël**

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

```
commit 27f2c4f2a8af19aee0fad71c787debbc440e67d1
Author: Thomas NOEL <tnoel@entrouvert.com>
Date:   Wed Feb 6 00:11:01 2019 +0100
```

```
add test tools in formdata inspect view (#23685)
```

**#19 - 07 février 2019 18:15 - Frédéric Péters**

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

**#20 - 04 mars 2019 19:42 - Frédéric Péters**

- Lié à *Development #17345*: zone de test d'expression sur la page */inspect* ajouté

**#21 - 04 mars 2019 20:39 - Frédéric Péters**

- Duplique *Development #14225*: dans les ateliers *formulaire/workflow* : avoir un testeur d'expression ajouté

**Fichiers**

---

Screenshot_2019-02-06 Backoffice de Démarches - vide.png	41,3 ko	05 février 2019	Thomas Noël
0001-add-test-tools-in-formdata-inspect-view-23685.patch	13,2 ko	06 février 2019	Thomas Noël
0001-add-test-tools-in-formdata-inspect-view-23685.patch	13,7 ko	06 février 2019	Thomas Noël