

## w.c.s. - Development #36924

### enregistrer les erreurs de calcul des conditions de page

15 octobre 2019 10:09 - Thomas Noël

<b>Statut:</b>	Fermé	<b>Début:</b>	15 octobre 2019
<b>Priorité:</b>	Bas	<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>	Non
<b>Patch proposed:</b>	Non		
<b>Description</b>			
<p>Historiquement, les erreurs de calcul sur les condition de page (condition d'affichage) ont le droit de complètement planter (par exemple si form_var_truc n'existe pas) ; dans ce cas la page ne s'affiche pas.</p> <p>On logue mais on n'enregistre cependant rien qui soit visible par l'admin fonctionnel au niveau du formulaire (comme les autres erreurs de calcul dans les workflow, par exemple).</p> <p>Ca pourrait sans doute être aujourd'hui re-discuté, c-à-d qu'on pourrait enregistrer les erreurs ; notamment parce que les conditions sont par défaut en Django, moins "plantant".</p>			

#### Historique

##### #2 - 10 novembre 2023 17:26 - Frédéric Péters

- Statut changé de Nouveau à Fermé

C'est désormais le cas, involontairement.

On a une méthode `evaluate_condition()`, qui appelle `PageCondition(...).evaluate()`.

`PageCondition` est une classe qui hérite de la classe `Condition`, la différence qui nous intéresse est qu'elle désactive l'enregistrement des erreurs :

```
class PageCondition(Condition):
    record_errors = False
```

Mais avec [#76480](#) pour ne pas enregistrer d'erreurs pour les testdef, il y a eu ça :

```
- def evaluate_condition(dict_vars, formdef, condition):
-     return PageCondition(condition, {'dict_vars': dict_vars, 'formdef': formdef}).evaluate()
+ def evaluate_condition(dict_vars, formdef, condition, record_errors=True):
+     return PageCondition(
+         condition, {'dict_vars': dict_vars, 'formdef': formdef}, record_errors
+     ).evaluate()
```

et donc l'appel dans le cas des conditions de sortie de page va avoir `record_errors` à `True`, qui va être passé au constructeur de `PageCondition`, et qui va écraser la valeur de l'attribut qui était `False`.

C'est ainsi depuis plus de six mois et ça n'a rien soulevé à ma connaissance, donc tout va bien.