

Publik - Development #27429

Permettre les champs readonly

19 oct. 2018 11:34 - Laurent Séguin

Statut:	Information nécessaire	Début:	19 oct. 2018
Priorité:	Normal	Echéance:	
Assigné à:		% réalisé:	0%
Catégorie:	Fabriques et traitement (wcs)	Temps estimé:	0.00 heure
Version cible:	2019	Demande du club utilisateur:	Oui
Patch proposed:	Non		

Description

En tant qu'administrateur fonctionnel je veux verrouiller certains champs de mon formulaire afin d'empêcher l'utilisateur d'en modifier la valeur.

// Lister tous les cas d'usage connus

Historique

#1 - 19 oct. 2018 20:55 - Thomas Noël

J'ai pas compris. Un champs que l'utilisateur ne peut pas modifier, il restera vide... quel intérêt ?

#2 - 19 oct. 2018 23:25 - Benjamin Dauvergne

Thomas Noël a écrit :

J'ai pas compris. Un champs que l'utilisateur ne peut pas modifier, il restera vide... quel intérêt ?

Ça suppose en plus que le champ est pré-rempli.

#3 - 20 oct. 2018 09:24 - Frédéric Péters

Ça suppose plein de choses, il y a une série de situations, avec des réponses différentes, qui se trouvent derrière ce ticket. C'est l'objet du "lister tous les cas d'usage connus", sans quoi le ticket est creux, oui.

En y allant avec mon biais technique, il y a des questions d'affichage, pour éviter le hack de créer un champ liste avec un seul élément.

Mais il n'y a pas juste "lecture seule", il y a aussi "caché", la pratique existe d'utiliser un champ de formulaire prérempli par une expression Python pour gagner une valeur calculée, utilisable plus tard. Avec du hack pour taper le champ en "display: none", ça donne par exemple [#18911](#). Ou l'utilisation de `session_var_` qui cassent lors de saisies concurrentes de demandes parce que l'info n'est pas attachée à une demande.

#4 - 20 oct. 2018 11:54 - Stéphane Laget

Frédéric Péters a écrit :

C'est l'objet du "lister tous les cas d'usage connus", sans quoi le ticket est creux, oui.

(...) Ou l'utilisation de `session_var_` qui cassent lors de saisies concurrentes de demandes parce que l'info n'est pas attachée à une demande.

Exemple à Villeurbanne :

Traitement des demandes d'emplois.

Les postes et leur descriptifs sont exposés sur le site de la Ville, le bouton "postuler" renvoie vers Publik avec plusieurs variables de session, dont le libellé du poste (mais également d'autres infos comme la date de réponse minimum et le service en charge du recrutement)

On a besoin du nom du poste dans le formulaire.

Le pré-remplissage d'un champ form_var_poste avec la session_var_ permet de disposer du nom du poste dans une variable de formulaire, et est enregistrée tout de suite. Mais pour cela, il faut soit cacher le champ au candidat, soit lui exposer ce champ en l'empêchant de le modifier (read-only) Si le candidat ne va pas au bout et décide de garder sa candidature dans un brouillon, il peut reprendre sa demande plus tard, avec le nom du poste qui sera enregistré dans form_var_poste.

Sans le pré-remplissage de cette variable, s'il reprend son brouillon, la variable de session est perdue, et le candidat répond à une offre d'emploi sans que l'on sache à quel poste il répond !

Aujourd'hui cela se contourne en cachant le champ au candidat avec la classe "hidden" mais ce n'est pas satisfaisant.

#5 - 22 oct. 2018 15:09 - Pierre Cros

- Demande du club utilisateur mis à Oui

#6 - 06 nov. 2018 07:36 - Frédéric Péters

- Statut changé de Nouveau à Information nécessaire

"Information nécessaire" pour "lister tous les cas d'usage connus" de la description.

#7 - 21 nov. 2018 08:23 - Victor Claudet

Saisi d'un point sur un champ carte.

Ce point permet de pré-remplir des champs adresse (numéro, voie, code postal ville) et ces champs devraient ne pas être modifiables (en particulier dans les cas où le point de carto entraîne des automatismes dans le circuit de traitement ensuite => découpage par quartier, responsable de traitement différent suivant la zone de déclaration...)

#8 - 21 nov. 2018 12:02 - Benjamin Dauvergne

Nanterre / tous les projets qui font des formulaires BO liés à un identifiant externe (dossier, fiches, que sais-je)

Pouvoir persister un paramètre passé en session et le référencer pendant le remplissage du formulaire (appels à la variable webservice notamment) et plus tard dans le workflow, permettre d'ouvrir plusieurs fois le même formulaire avec un paramètre différent (les session_var s'écrasent si même paramètre utilisé dans deux onglets avec des valeurs différentes).

(rajout de Thomas : en l'occurrence ça serait ici des champs read-only + remplis + cachés, genre "form_var_rsu_id" où on stocke "session_var_rsu_id" -- pour l'instant on le fait dans l'étape 1 du workflow dans une donnée de traitement, avec le risque que les session_var aient été modifiées par un autre formulaire dans un autre onglet entre temps)

#9 - 21 nov. 2018 12:17 - Stéphane Laget

Cas d'usage à Grenoble :

Une demande "complexe" d'autorisation de manifestation qui doit être faite en 2 temps :

- une pré-demande (avec instruction sur la faisabilité et l'opportunité)
- un demande complète (= dossier technique en lien avec la pré-demande)

La 2eme demande est automatiquement remplie par les éléments de la 1ère demande. On peut créer cette 2ème demande avec l'action de re-soumission, ou un webservice interne pour créer une 2ème demande, ou l'utilisation d'un JDS ou en restant sur la 1ère demande et en jouant avec des pages conditionnelles pilotées par une donnée de traitement.

Mais dans tous les cas, cela permet de pré-remplir des champs qui restent modifiables par l'utilisateur, alors que l'on ne voudrait pas, car ces éléments ont été instruits et validés par l'agent lors de la pre-demande.

Si ces champs ne sont pas affichés dans le formulaire à l'utilisateur (parce que cachés dans une page conditionnelle), actuellement on les perd lors de la soumission du formulaire par l'utilisateur.

L'option "read-only" sur les champs réglerait le pb.