w.c.s. - Bug #7619

sso silencieux en cas d'erreur

18 juin 2015 09:20 - Frédéric Péters

Statut:	Nouveau	**Début:**	18 juin 2015
Priorité:	Normal	**Echéance:**	
Assigné à:		**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Non	**Planning:**	

Description

Un SSO avec l'idp qui refuse la connexion, wcs (auquo) n'affiche rien comme message.

Historique

#1 - 18 juin 2015 10:28 - Benjamin Dauvergne

Oui c'est le cas RequestDenied que jusqu'à présent on considérait uniquement comme le signe que l'utilisateur avait cliqué sur "Cancel" mais maintenant que l'on fait de l'autorisation coté IdP ce ne sera plus le cas. À noter qu'il y a un champ trop peu usité login.response.status.statusMessage qui permet à l'IdP de transmettre une phrase localisé à afficher par le SP; authentic ne le fait pas mais c'est à utiliser plus.

J'utiliserai bien get_session.message / display_message() comme je le fais avec django.contrib.messages mais à faire un git grep display_message() je ne vois rien dans un fichier qui semblerait s'afficher aux usagers (je ne vois que des références dans admin/backoffice ou ident/password.py).

```
admin/fields.py:        r += get_session().display_message()
admin/forms.py:        r += get_session().display_message()
admin/forms.py:        r += get_session().display_message()
admin/forms.py:        r += get_session().display_message()
admin/forms.py:        r += get_session().display_message()
admin/settings.py:            r += get_session().display_message()
admin/workflows.py:        r += get_session().display_message()
admin/workflows.py:        r += get_session().display_message()
admin/workflows.py:        r += get_session().display_message()
backoffice/management.py:        r += get_session().display_message()
qommon/ident/password.py:        r += get_session().display_message()
qommon/ident/password.py:        r += get_session().display_message()
qommon/ident/password.py:        r += get_session().display_message()
```

C'est utilisable en frontoffice à ton avis ?

#2 - 18 juin 2015 10:39 - Frédéric Péters

C'est pas des masses intégré, même en backoffice; ça devrait sans doute créer un '<ul class=messages>' comme dans django, qui serait posé dans une variable messages dans wcs/qommon/template.py:decorate(), puis adaptation des thèmes.

Bref, je dirais que oui, c'est l'infra à utiliser ici, et ça peut passer par un simple display_message() posé sur l'accueil, et puis dans un autre temps le travail dont je parle avant (et pas d'urgence et peut-être qu'on arrivera à merger wip/django avant).