

Authentic 2 - Development #41919

Permettre la saisie d'adresse en autocomplétion lors de la création et modification de compte

21 avril 2020 12:08 - Stéphane Laget

Statut:	Fermé	Début:	21 avril 2020
Priorité:	Normal	Echéance:	30 octobre 2020
Assigné à:	Lauréline Guérin	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposé:	Oui		
Description Sur le modèle de ce qui a été fait pour wcs, permettre ce fonctionnement lors des procédures de création et de modification de compte. L'échange sur la liste : Coté implémentation de l'IHM je comptais qu'on attende ce qui est fait coté w.c.s et recopier ça a la sauce Django, ça évitera d'avoir à réfléchir. Reste le problème du stockage et de la diffusion, l'idéal pour configurer ça ce serait que ce soit un type unique de champ (adresse) mais derrière je n'ai pas d'idée sur comment l'exploiter, pour l'instant nos attributs ne sont que des chaînes (vu des autres briques, dans A2 on peut stocker du JSON). Soit on étend le support des attributs (w.c.s., django-mellon, etc..) pour pouvoir décrire éventuellement au niveau SAML que c'est du JSON : <saml:AttributeValue xsi:type="eo:json">...</saml:AttributeValue> c'est un poil ad-hoc mais bon, sinon imaginer une implémentation qui derrière un unique attribut déclaré génère plusieurs attributs réel au niveau d'authentic (adresse -> adresse_rue, adresse_numero, etc..) ça fait plein de code dans authentic beaucoup moins ailleurs (enfin il en faudra dans hobo pour configurer la diffusion SAML de cet attribut complexe).			
Demandes liées: Lié à Intégrations graphiques Publik - Development #47474: Suppression du pre... Fermé 08 octobre 2020			

Révisions associées

Révision 7b130d6f - 09 octobre 2020 14:15 - Lauréline Guérin

profile_views: address autocomplete field (#41919)

Historique

#1 - 22 juin 2020 11:08 - Emmanuel Cazenave

- Assigné à mis à Emmanuel Cazenave

#2 - 22 juin 2020 11:09 - Benjamin Dauvergne

Ignorer mes considérations sur le transfert de l'adresse comme un tout.

#5 - 30 septembre 2020 10:52 - Brice Mallet

- Echéance mis à 30 octobre 2020

#6 - 01 octobre 2020 16:40 - Lauréline Guérin

- Assigné à changé de Emmanuel Cazenave à Lauréline Guérin

#7 - 02 octobre 2020 14:44 - Frédéric Péters

Benjamin peut préciser ses préférences par rapport à la description du ticket, ou peut-être Manu avait déjà regardé ça et vu la direction à prendre.

De mon côté je serais à fuir les modifications de fond et mimer au maximum l'existant w.c.s. : commencer par ajouter un type de champ, genre "autocomplétion d'adresse", qui correspondrait à ce qu'on a dans w.c.s. via

https://git.entrouvert.org/wcs/git/tree/wcs/qommon/templates/qommon/forms/widgets/select_jsonp--address.html : un select2 + un input

type=checkbox; le select2 serait configuré pour aller chercher les adresses via une URL tapée en settings.

Ensuite le js similaire à <https://git.entrouvert.org/wcs.git/tree/wcs/qommon/static/js/qommon.geolocation.js#n58>, i.e. on('change'...) qui sur la sélection d'une valeur via le select2 va remplir les champs address/zipcode/city/country (ces noms étant simplement en dur dans le js).

Sur la vue en édition, la même chose, juste en remplissant en amont le select2 avec une adresse reconstituée à partir des données des autres champs.

Ça laisse juste côté w.c.s. à ignorer le nouveau type de champ; ça évite d'avoir à concevoir un nouveau système de champ/donnée complexe dans authentic qui aurait quand même à conserver la compatibilité avec l'existant.

Voilà pour l'approche que j'aurais, qui me semble le meilleur moyen d'obtenir rapidement un résultat ici.

#8 - 02 octobre 2020 17:57 - Benjamin Dauvergne

Je n'ai pas d'avis sur l'implémentation, si ça marche pour w.c.s. c'est bon ici aussi (à voir si on peut mutualiser le fichier js ou pas).

Juste sur le point d'ignorer le nouveau type de champ coté w.c.s., dans le cas de Toulouse on voudra stocker l'id (code STI) qu'il retourne, ça me va de se dire qu'on conserve aussi le code BAN ou autre dans les autres cas même si on ne s'en sert pas actuellement.

#9 - 05 octobre 2020 10:45 - Lauréline Guérin

- Tracker changé de Support à Development

#10 - 06 octobre 2020 15:49 - Lauréline Guérin

- Fichier 0001-profile_views-address-autocomplete-field-41919.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

#11 - 06 octobre 2020 16:08 - Frédéric Péters

(juste un premier survol), parce que c'est un truc que j'avais noté avant, pour :

```
+ $('#id_address').val(number_and_street);
```

sur la page publique de modification du profil (/accounts/edit/), le tag a un id différent, "id_edit-profile-address"; je ne sais pas si ça a un sens et que ça gagnerait juste à être modifié pour avoir partout le même id, ou s'il vaut mieux ajouter un attribut supplémentaire, genre data-attribute-name="address".

#12 - 06 octobre 2020 17:09 - Benjamin Dauvergne

Frédéric Péters a écrit :

sur la page publique de modification du profil (/accounts/edit/), le tag a un id différent, "id_edit-profile-address"; je ne sais pas si ça a un sens et que ça gagnerait juste à être modifié pour avoir partout le même id, ou s'il vaut mieux ajouter un attribut supplémentaire, genre data-attribute-name="address".

On peut retirer le préfixe, il du servir dans le passé ou il était prévu qu'il serve mais ce n'est plus le cas. Par contre ça fait quelques tests à modifier à coup de sed (et le edit-profile-next-url aussi).

```
src/authentic2/views.py:136:         kwargs['prefix'] = 'edit-profile'
src/authentic2/views.py:144:         field_name='edit-profile-next_url',
tests/test_attribute_kinds.py:459:     form.set('edit-profile-first_name', 'John')
tests/test_attribute_kinds.py:460:     form.set('edit-profile-last_name', 'Doe')
tests/test_attribute_kinds.py:461:     form.set('edit-profile-cityscape_image-clear', True)
tests/test_attribute_kinds.py:473:     form.set('edit-profile-cityscape_image', Upload('tests/201x201.jpg'))
tests/test_attribute_kinds.py:483:     assert form['edit-profile-cityscape_image'].attrs['accept'] == 'image/*'
tests/test_profile.py:50:     resp.form['edit-profile-phone'] = '1234'
tests/test_profile.py:51:     assert resp.form['edit-profile-phone'].attrs['type'] == 'tel'
tests/test_profile.py:52:     resp.form['edit-profile-title'] = 'Mrs'
tests/test_profile.py:53:     resp.form['edit-profile-agreement'] = False
tests/test_profile.py:73:     resp.form.set('edit-profile-phone', '0123456789')
tests/test_profile.py:78:     resp.form.set('edit-profile-phone', '9876543210')
tests/test_profile.py:86:     assert 'edit-profile-phone' not in resp.form.fields
tests/test_profile.py:87:     assert 'edit-profile-title' not in resp.form.fields
tests/test_profile.py:88:     assert 'edit-profile-agreement' not in resp.form.fields
tests/test_profile.py:89:     assert 'readonly' in resp.form['edit-profile-phone@disabled'].attrs
tests/test_profile.py:90:     assert resp.form['edit-profile-phone@disabled'].value == '0123456789'
tests/test_profile.py:91:     assert resp.form['edit-profile-title@disabled'].value == 'Mr'
tests/test_profile.py:92:     assert resp.form['edit-profile-agreement@disabled'].value == 'Yes'
tests/test_profile.py:93:     resp.form.set('edit-profile-phone@disabled', '1234')
tests/test_profile.py:94:     resp.form.set('edit-profile-title@disabled', 'Mrs')
```

```

tests/test_profile.py:95:      resp.form.set('edit-profile-agreement@disabled', 'False')
tests/test_profile.py:109:     assert 'edit-profile-phone@disabled' not in resp
tests/test_profile.py:110:     assert 'edit-profile-title@disabled' in resp
tests/test_profile.py:111:     assert 'edit-profile-agreement@disabled' in resp
tests/test_profile.py:125:     resp.form.set('edit-profile-phone', '0123456789')
tests/test_profile.py:131:     resp.form.set('edit-profile-phone', '1234')
tests/test_profile.py:156:         return set(key.split('edit-profile-')[1]
tests/test_profile.py:157:                     for key in resp.form.fields.keys() if key and key.startswith('edit-profile-'))
tests/test_profile.py:188:     assert len(response.pyquery('input[type="radio"][name="edit-profile-title"]')) == 2
tests/test_profile.py:189:     assert len(response.pyquery('input[type="radio"][name="edit-profile-title"][readonly="true"]')) == 0
tests/test_profile.py:190:     assert len(response.pyquery('select[name="edit-profile-title"]')) == 0
tests/test_profile.py:195:     assert len(response.pyquery('input[type="radio"][name="edit-profile-title"]')) == 0
tests/test_profile.py:196:     assert len(response.pyquery('input[type="text"][name="edit-profile-title@disabled"][readonly]')) == 1
tests/test_registration.py:382:    assert 'edit-profile-profession' in response.form.fields
tests/test_registration.py:383:    assert 'edit-profile-prenom' not in response.form.fields
tests/test_registration.py:384:    assert 'edit-profile-nom' not in response.form.fields
tests/test_registration.py:386:    assert response.pyquery('[for=id_edit-profile-profession]')
tests/test_registration.py:387:    assert not response.pyquery('[for=id_edit-profile-profession].form-field-required')
tests/test_registration.py:388:    response.form.set('edit-profile-profession', 'pompier')

```

#13 - 06 octobre 2020 19:10 - Benjamin Dauvergne

Et j'oubliais les thèmes :

```

static/includes/_forms.scss:div.a2-block ul#id_edit-profile-title {
static/minint/initial.css:#id_edit-profile-email_wrap, #id_edit-profile-address_wrap {
static/minint/initial.css:#id_edit-profile-first_name_wrap, #id_edit-profile-phone_wrap, #id_edit-profile-postal_code_wrap {
static/minint/initial.css:#id_edit-profile-last_name_wrap, #id_edit-profile-mobile_wrap, #id_edit-profile-city_wrap {
static/strasbourg-2018/_custom.scss:#id_edit-profile-title li {

bdauvergne@revestel:~/wd/eo/cut-publik-theme$ git grep edit-profile
static/grandlyon-cut/_custom.scss:    ul#id_edit-profile-title li, #registration_completion li {
static/grandlyon-cut/_custom.scss:    ul#id_title, ul#id_edit-profile-title {

```

#14 - 08 octobre 2020 11:59 - Lauréline Guérin

- Fichier 0001-profile_views-address-autocomplete-field-41919.patch ajouté

Pour authentic, je regarde les thèmes dans la foulée.

Pour la page accounts/edit, ça donne quelque chose d'un peu étrange si l'un des champ adresse est setté avec le flag verified=True
Est-ce qu'on prévoit quelque chose de particulier pour ce cas ?

#15 - 08 octobre 2020 13:02 - Benjamin Dauvergne

Lauréline Guérin a écrit :

Pour authentic, je regarde les thèmes dans la foulée.

Pour la page accounts/edit, ça donne quelque chose d'un peu étrange si l'un des champ adresse est setté avec le flag verified=True
Est-ce qu'on prévoit quelque chose de particulier pour ce cas ?

TL;DR; ne pas s'en préoccuper ni tester ce cas dans ce ticket.

Le principe pour l'instant c'est qu'un champ éditable par l'utilisateur ne l'est plus si il est vérifié. La logique voudrait qu'on flag tous les champs adresse ; le champ autocomplete aussi, donc pas d'interaction entre un autocomplete actif et des champs inactifs. Dans les faits à part GLC qui gère ça totalement différemment on a aucun endroit avec des champs d'adresse vérifiés sauf ceux issus de FranceConnect et l'adresse n'en fait pas partie (indirectement sur Toodego mais le profil vient de GLC et la page d'édition du profil est désactivé).

#18 - 08 octobre 2020 14:18 - Lauréline Guérin

- Lié à Development #47474: Suppression du prefix edit-profile- ajouté

#19 - 09 octobre 2020 13:30 - Benjamin Dauvergne

- Statut changé de Solution proposée à Solution validée

#20 - 09 octobre 2020 15:18 - Lauréline Guérin

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

```
commit 7b130d6ffc34e1a6b706aa460f1e3ab7a32d5f2a
Author: Lauréline Guérin <zebuline@entrouvert.com>
Date: Tue Oct 6 15:44:54 2020 +0200
```

```
profile_views: address autocomplete field (#41919)
```

#21 - 13 octobre 2020 14:16 - Frédéric Péters

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

Fichiers

0001-profile_views-address-autocomplete-field-41919.patch	13,5 ko	06 octobre 2020	Lauréline Guérin
0001-profile_views-address-autocomplete-field-41919.patch	23,1 ko	08 octobre 2020	Lauréline Guérin