

Authentic 2 - Development #6919

Create a custom user model

08 avril 2015 15:21 - Benjamin Dauvergne

Statut:	Fermé	Début:	08 avril 2015
Priorité:	Haut	Echéance:	
Assigné à:	Benjamin Dauvergne	% réalisé:	100%
Catégorie:		Temps estimé:	0:00 heure
Version cible:	2.2.0	Planning:	
Patch proposed:	Non		
Description			
We should return to using a custom user model:			
<ul style="list-style-type: none">• username length is 256 chars• username is nullable and optional• email is 254 chars• email default validator is our own• User has an automatic uuid field using uuid.uuid4• Module fix_user_model.py can be removed• Old user models are automatically converted to new custom user model• Registration form does not have to generate an uuid username anymore			

Révisions associées

Révision 01d6a0c4 - 15 avril 2015 11:30 - Benjamin Dauvergne

Remove support for Django < 1.7 (#6919)

- call plugin initializations in authentic2 application's ready hook
- remove pre-Django-1.7 check before AppConfig classes
- remove pre-Django-1.7 JSON serialization module
- remove south from settings.py
- remove pre-Django-1.7 send_mail support in registration form
- remove pre-Django-1.7 support of TransactionMiddleware in setup.py
- remote pre-Django-1.7 support of the URLValidator validator

Révision 96500dd0 - 15 avril 2015 11:30 - Benjamin Dauvergne

Create a custom user model (fixes #6919)

- Add new model User through a custom user application, Django needs custom user models to be introduced in the first migration of an application, it features:
 - username length is 256 chars
 - username is nullable and optional
 - email is 254 chars
 - email default validator is our own
 - User has an automatic uuid field using uuid.uuid4
 - Module fix_user_model.py can be removed
 - Old user models are automatically converted to new custom user model
 - Registration form does not have to generate an uuid username anymore
- Add migrations to fix foreign key in our own applications and in third party applications out of our control like django-admin-tools and django-admin (the AlterField migration operation was overridden for these cases)
- django.contrib.auth.models.User to authentic2.models.User
- Set AUTH_USER_MODEL to authentic2.User
- Register custom_user.User in admin site
- Remove all monkey patching to Django default user model
- username field is now forced in the model backend, as the USERNAME_FIELD attribute is now `uuid`.
- Create a local modelform_factory implementation, it specializes User model by always using authentic2.forms.BaseUserForm as a base class and add a new parameter `required` which allow to set the required flag on fields without setting it on the model.
- Add a BaseUserForm base model-form class supporting custom fields created using the Attribute model.

- Refactor profile editing form, user admin form, manage user edit form and registrationform using the new modelform_factory based on the BaseUserForm class.
- Added new settings:
 - A2_USERNAME_IS_UNIQUE: it adds checking that the username attribute is unique before saving a user object
 - A2_REGISTRATION_USERNAME_IS_UNIQUE: it adds checking that the username attribute is unique before saving a user object; same as above but only for self-registered users.
 - A2_EMAIL_IS_UNIQUE: same as above but for email
- Remove all unused symbols in authentic2.compat: use_attribute_aggregator, get_required_fields and get_registration_fields
-

Historique

#1 - 15 avril 2015 11:31 - Benjamin Dauvergne

- Statut changé de Nouveau à Résolu (à déployer)
- % réalisé changé de 0 à 100

Appliqué par commit [96500dd0ac8ea6328ed93d26b51c42c66022844d](#).

#2 - 23 février 2016 12:58 - Benjamin Dauvergne

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

#3 - 06 décembre 2017 15:28 - Benjamin Dauvergne

- Statut changé de Solution déployée à Fermé