Annuaire des associations

L'annuaire des associations est la partie du portail qui recense les associations sous toutes leurs formes (Asso 1901, clubs, listes...)

Types de contenus

Premièrement, le type de contenu **association** a été crée avec les champs suivants :

Ici le module **Field Collection** et **Entity Reference** sont nécessaires au fonctionnement de ce type de contenu.

Champ	Type	Description
Nom	texte	Nom de l'asso
Présentation	texte	courte présentation de l'association
Logo	Image	Logo servant à identifier l'asso dans les listes
Membres	Field collection membre asso	La liste des membres, leurs poste et mandat
Mandat actuel	Entier	Année du mandat actuel, servira à afficher les membres par mandat

Pour la description du field collection membres :

Champ	Type
user	Référence à un utilisateur
poste	texte
mandat	entier

** TODO ** Ajouter un champ type (asso, club, liste...), des liens de parenté pour les sous-com...

Mise en place de l'affichage

Ici les modules **Display Suite** et **Display Suite UI** sont nécessaires.

Page d'affichage d'une association

On fait les modification sur l'affichage par défaut d'un contenu **association**.

Avec le module display suite, on utilise la mise en page *Deux colonnes fluides*. A gauche on met les informations relatives à l'asso (nom, logo..) et à droite la liste des membres. Le display de la liste des membres se fait avec le mode contenu complet, et avec un affichage *icone* que l'on définit directement dans la field collection membres.

Affichage des membres

L'affichage des membres va se faire de manière un peu spéciale étant donné qu'on a plusieurs contraintes :

- Il faut afficher uniquement les membres du mandat actuel
- Il faut pouvoir afficher les membres des mandats précédents

Etant donné que tout est stocké dans une field collection, et que les affichages par défaut ne permettent pas de faire de tri on va utiliser une **view**.

On commence donc par créer une view qui va afficher des éléments de *field collection*. On désactive *créer une page* et on active *créer un bloc* à la place. Il faut mettre un titre explicite, comme *membres assos* par exemple.

Pour ce qui est de la configuration de la vue on procède comme suit :

- **Titre de la vue :** Membres de l'association
- **Format :** liste non mise en forme, avec classe de ligne *col-md-4* (**TODO** Ajouter les classes pour un bon responsive)
- **Afficher :** Entité rendue
- **Critère de filtrage :** Élément field collection mandat (=2017) (exposé)
- **Nom du bloc :** membres assos
- **Pagination :** Afficher tous les éléments

Ensuite dans avancé on va rajouter toute la magie [1)] qui va faire en sorte que la vue fasse bien son job : Sur chaque page d'association afficher uniquement les membres de l'asso consultée. (**Dans l'ordre indiqué sinon ça marche pas**)

- **Relations :** Ajouter Élément Field collection : Entité avec membres (field_membres)
- **Filtres contextuels :** Contenu : nid
  - **Relation :** field_membres
  - **Quand la valeur de filtre N'EST PAS disponible :** Fournir une valeur par défaut (Type : Identifiant (ID) du contenu à partir de l'URL)

Ensuite on revient à l'affichage du contenu association. On va créer un champ personnalisé qui va être un *champ bloc*. Celui-ci va bien-sûr faire appel à la view que l'on vient de configurer ! On peut lui donner un nom du genre *membres_render*.

Ensuite il reste à gérer la façon dont les membres seront affichés. Pour ça on va directement gérer l'affichage de la field collection *field_membres*

L'affichage se fait avec un champ personnalisé *User icon* (c'est le petit nom qu'on lui donne) et on va coder directement l'affichage. (Un peu de code ça fait pas de mal)

Pour cela il faut activer display suite, en mettant un champ d'affichage default par exemple. Ensuite on va dans le menu champs personnalisé dans le menu du bas, et on ajoute un champ de code.

```
<div class="well" style="text-align:center;">
[field_collection_item:field-user:field_portrait]
[field_collection_item:field-user:field_prenom]
[field_collection_item:field-user:field_nom]
```

```
</div>
```



Pour que ça marche bien :

- Il faut que les tokens soient activés (c'est ce qui permet d'inclure dynamiquement du texte)
- Il faut que dans les paramètres d'affichage utilisateur, dans la section Token, pour le champ portrait l'affichage de l'image se fasse avec le style icone
- Si le style icone n'est pas crée : Recadrer et redimensionner (150x150px)

TODO Virer ce CSS inline et rajouter une classe pour le mettre en injection CSS.

1)

Comprendre que j'ai passé des heures à trouver comment faire

From:
https://wiki.centrale-med.fr/ginfo/ - **Wiki GInfo**

Permanent link:
https://wiki.centrale-med.fr/ginfo/webmastering:portail:assos

Last update: **16/07/2017 17:57**