

Intégrations graphiques Publik - Bug #49878

rendre disponible le CSS de links-list--link à d'autres liens

05 janvier 2021 11:29 - Thomas Jund

Statut:	Rejeté	Début:	05 janvier 2021
Priorité:	Normal	Echéance:	
Assigné à:	Thomas Jund	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		

Description

Suite à demande de Steph #49116 : pouvoir appliquer le design des liens d'une "link list" à la cellule "lien vers une démarche" (wcs-form-cell)

l'idée première serait de pouvoir simplement appliquer le style via un @extend sass :

```
.wcs-form-cell a {
  @extend .links-list--link;
}
```

Historique

#1 - 05 janvier 2021 11:47 - Thomas Jund

- Fichier 0001-scsss-make-extendable-links-list-link-styles-49878.patch ajouté
- Tracker changé de Support à Bug
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

Patch

en isolant les styles dans une class `links-list-link` on gagne aussi lors de la création de thème. Pour modifier le design des liens, terminé avec la longue liste de sélecteurs.

AVANT:

```
div#rub_service div.category ul,
div#services > ul > li > ul,
div#account-management ul,
div.links-list ul,
div.menuecell ul,
div.wcsformcell,
div.categoriescell ul,
div.wcscurrentdraftscell ul,
div.wcscurrentformscell ul {
  & > li > a {
    // styles en plus
  }
}
```

AVEC PATCH

```
.links-list--link {
  // styles en +
}
```

Ce qui peut grandement améliorer les supports / évolutions du code.

En bonus dans un futur possible (demande avis)

Il est possible de définir une variable sass qui appliquerait le layout d'un composant, il faut passer par une mixin mais possible

```
@mixin apply-style-of($selector) {
  @extend #{$selector};
}
```

```
$wcs-form-cell-link-styles: null // .links-list--link , .pk-button
```

```
@if $wcs-form-cell-link-styles {  
  .wcs-form-cell a {  
    @include apply-style-of($back-top-link-extend);  
  }  
}
```

et donc en fixant `\$wcs-form-cell-link-styles: .pk-button` on fait hériter le style .pk-button à ce lien, intéressant aussi pour l'interface de création de thème.

#3 - 05 janvier 2021 11:53 - Frédéric Péters

Pourquoi ne pas faire évoluer l'HTML pour partout ajouter une classe links-list ? Là on se trouve à la fois avec toutes la série de sélecteur ET le truc tiret tiret; et quand c'est l'ul, ou le li, et pas le a, qu'on voudra stylé, ça repartira pour un tour.

#4 - 05 janvier 2021 12:09 - Thomas Jund

Il me semble que c'est exactement vers cela que ce patch nous emmène. L'objectif de réduire la liste des sélecteurs est aussi prévue. L'un n'empêchant pas du tout l'autre ou je n'ai pas compris tes remarques. Tu souhaiterais pousser le patch plus loin ?

#5 - 05 janvier 2021 12:31 - Frédéric Péters

Cette étape qui amène une situation avec à la fois la série de sélecteurs ET le tiret-tiret se présente pour moi comme une régression. J'aurais préféré que la classe links-list soit ajoutée partout, réduire la multiplicité des sélecteurs, avant de partir sur quelque chose de nouveau.

#6 - 05 janvier 2021 12:39 - Frédéric Péters

L'objectif de réduire la liste des sélecteurs est aussi prévue.

Bref je préférerais que ça commence par là, concrètement.

#7 - 06 janvier 2021 08:09 - Frédéric Péters

À revenir au début, par-delà les commentaires sur le patch, c'est l'idée même du ticket qui me pose problème : si on veut une liste de liens, il y a une cellule liste de liens, et possibilité qu'il y ait dedans un seul lien, et il aura le style d'un lien dans une liste de liens.

(ça n'empêche pas l'utilité qu'il y aurait à réduire la liste des sélecteurs)

#8 - 06 janvier 2021 10:06 - Thomas Jund

- Statut changé de Solution proposée à Rejeté

si on veut une liste de liens, il y a une cellule liste de liens

Mazette mais bien sûr ! Je m'en vais répondre cela à Steph.

Au-delà de ce besoin précis, J'aime quand même bien les idées qui introduisent ce patch comme ouvrir la possibilité d'appliquer un modèle parmi plusieurs à un composant sans venir en surcouche. Bien plus propre à mon avis qu'un slug qui nécessite encore trop souvent une réécriture spécifique du markup de la cellule (alourdi la maintenance) ou l'utilisation d'une class additionnelle qui devra d'abord reseter les styles existants par défaut et nécessite d'être maintenu séparément du reste du code.

Dans la logique d'un design system modulaire, définir un style et un comportement d'un côté et ensuite pouvoir choisir dans un thème quel élément hérita de ce style apporte beaucoup de souplesse.

Définir d'un côté le design d'un lien (typo, couleur, survol, etc.) et dire : applique ce style par défaut aux liens des listes de liens, mais aussi aux cellules "lien vers une démarche" et "lien vers une catégorie" permet d'être bien plus efficace que de devoir réécrire du code spécifique qui nous entrave ensuite.

En matière d'UX, permettre à un agent (et nos CPFs) de créer leur page en utilisant une cellule "lien vers une démarche" parce que c'est simplement son besoin et que ça marche et pas de devoir assimiler que pour faire un lien unique vers une démarche il ne faut pas utiliser "Lien vers une démarche" car l'affichage est incorrecte mais qu'il faut utiliser une cellule "liste de liens" sans mettre de titre a son importance.

Je laisse mûrir un peu l'idée.

Fichiers

0001-scss-make-extendable-links-list-link-styles-49878.patch	3,47 ko	05 janvier 2021	Thomas Jund
--	---------	-----------------	-------------