



Guide de rédaction des créations HTML

Révisions

Révision	Auteur	Date	Commentaire
1	ganne	08/04/2015	Initialisation du guide
2	mlemaitre	09/04/2015	Relecture et commentaires
3	mlevallois	09/04/2015	Ajout de différentes informations
4	mlevallois	19/05/2015	Images
5	ganne	02/06/2015	MAJ pour l'import multipage
6	mlevallois	16/06/2015	MAJ du formulaire
7	ganne	29/07/2015	Suppression input fid dans exemple de form
8	mlevallois	07/08/2015	Ajout du store locator
9	ganne	05/04/2016	Nettoyage des " erronées, remplacées par des "
10	pmarechal	22/06/2016	Mise en forme / Nouvelle identité
11	ganne	10/04/2017	MAJ pour le module "Réseaux sociaux"
12	ganne	19/06/2017	Ajout de l'attribut uneditable + Fix detail form
13	srenouf	03/01/2018	Précisions sur la gestion des erreurs dans un formulaire
14	srenouf	16/05/2019	Ajout d'une partie "Problèmes d'affichages sur outlook"
14	pelmenjra	7/04/2020	Ajout d'une section pour décrire comment ignorer des champs de formulaire
15	srenouf	30/09/2020	Ajout de précisions sur l'import de fichiers et autres

Diffusion: public

Destinataire	Date	Commentaire
Marketing	08/04/2015	Diffusion pour validation

Table des matières

Contexte

1. Intégration dans l'éditeur

1.1. Import des créations dans l'application

1.1.1. Types de fichier

1.1.2. Email

1.1.3. Smartpage

1.2. Blocs éditables dans le HTML

1.2.1. Texte

1.2.2. Image

1.2.3. Map

1.2.4. Form

1.2.5. Vidéo

1.2.6. Social

1.2.7. Locator

1.3. Blocs déplaçable et supprimable

2. Contraintes à respecter

2.1. Entête HTML

2.2. Lien de désabonnement

2.3. Lien de preview

3. Astuces résolution incohérences d'affichages sur outlook

3.1. Propriétés supplémentaires outlook

3.2. Corrections d'affichage

3.2.1. Règles générales

3.2.2. Espaces supplémentaires

3.2.3. Images

3.2.4. Liens

3.2.5. Texte de droite à gauche

Contexte

Ce document a pour but de guider un utilisateur ou un prestataire afin de lui permettre de concevoir ses créations HTML (email ou smartpage) de façon autonome.

1. Intégration dans l'éditeur

Pour une intégration parfaite dans l'éditeur, nous conseillons de commencer votre email directement dans l'éditeur Digitaleo. Puis de récupérer le code HTML et le modifier sur votre éditeur de code. Ceci est facultatif mais vivement conseillé si vous souhaitez modifier la structure de votre email dans l'éditeur.

1.1. Import des créations dans l'application

1.1.1. Types de fichier

Il est possible d'importer deux types de fichiers :

- un fichier HTML simple contenant des images pointant vers un hébergement externe
- un archive ZIP contenant le(s) fichier(s) HTML dont un nommé **index.html** ainsi que les images pointées en relatif dans les pages HTML vers le dossier images. Le lien doit obligatoirement se présenter comme ceci : "images/logo.png"

L'encodage des fichiers HTML doit être le suivant :

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

1.1.2. Email

Pour importer un email vous pouvez fournir un fichier ou une archive ZIP contenant le fichier **index.html** ainsi que les images pointées en relatif dans les pages HTML

1.1.3. Smartpage

Pour importer une smartpage vous pouvez fournir un fichier **index.html** pour une smartpage **mono-page** ou une archive ZIP contenant des fichiers HTML (tous dans le même répertoire) dont au moins un nommé **index.html** ainsi que les images pointées en relatif dans les pages HTML.

Dans le cas d'une smartpage **multi-page** les liens entre les pages sont gérés de façon relative.

Exemple :

index.html

```
...  
<ul>  
  <li><a href="formulaire.html">Inscription</a></li>  
  <li><a href="map.html">Nous trouver</a></li>  
</ul>  
...
```

fomulaire.html

```
...
<form form-editor action="congrat.html" role="form">
  ...
  <button type="submit" class="btn btn-default center">Submit</button>
</form>

<a href="index.html">Home page</a>
...
```

map.html

```
...

<a href="index.html">Home page</a>
...
```

confirmation.html

```
...

<div text-editor>
  Félicitation
</div>
<a href="index.html">Home page</a>
...
```

1.2. Blocs éditables dans le HTML

1.2.1. Texte

Il est possible de définir une portion de texte éditable en ajoutant l'attribut *text-editor* sans valeur

Exemple de code

```
<div text-editor="">
  <p>Title</p>
  <p>Baseline</p>
  <p>Your content here</p>
</div>
```

Il est conseillé de ne pas mettre l'attribut directement sur un bloc P car cela limite les fonctionnalités d'édition

1.2.2. Image

Il est possible de rendre une image éditable (au sens interchangeable avec le contenu de la bibliothèque de contenu) en ajoutant l'attribut *img-editor* sans valeur sur une balise IMG ou sur une balise A contenant une balise IMG

Que votre code utilise l'une ou l'autre syntaxe, l'éditeur permet d'ajouter ou de supprimer un lien sur l'image.

Vous pouvez importer des images en même temps que votre code HTML, en important une archive ZIP. Cela vous permet d'utiliser des chemins relatifs pour vos images.

Vous pouvez aussi utiliser une image hébergée à l'extérieur dans votre code HTML, elle sera remplaçable via l'éditeur mais vous ne pourrez pas la réutiliser dans l'éditeur.

Les images peuvent être redimensionnées automatiquement, lorsque vous en sélectionnez une, s'il y a un `ie-resize="longueurxlargeur"`. Le redimensionnement ne prend effet que si les tailles sont renseignées en pixels. Le style et les classes de la balise image sont bien évidemment récupérés.

Exemple de code :

```

```

```
<a img-editor="" href="DEFAULT_TARGET">  
    
</a>
```

Attention : Les balises `<map>` et `<area>` permettant de créer des zones cliquables sur des images ne sont pas supportées. Nous vous conseillons de découper votre image en plusieurs images sur lesquelles vous pouvez ajouter un lien.

1.2.3. Map

Il est possible d'ajouter une carte statique relative à une adresse. Pour cela il faut utiliser une balise `IMG` avec l'attribut `map-editor`

Exemple de code

```

```

Astuce : Il est préférable de définir une source par défaut pour que l'aperçu avant édition soit correct (certains navigateurs n'affichent rien si un `img` n'a pas de source)

1.2.4. Form

Il est possible de définir un formulaire de saisie permettant de créer une liste de contact réutilisable dans l'outil. Pour cela il faut utiliser une balise `DIV` avec l'attribut `form-editor` et ensuite une balise `FORM`. Le titre du formulaire doit se trouver entre des balises `h2` (ainsi qu'un sous-titre entre des `h3`).

Nous conseillons cependant d'utiliser le constructeur de formulaire dans l'éditeur, puis de coller le code dans

votre éditeur de code. Le formulaire sera re-généré après chaque modification de celui-ci dans l'éditeur, écrasant toute modification dans le HTML.

Exemple de code généré :

```
<div form-editor="">
  <h2>Title</h2>
  <h3>SubTitle</h3>
  <div class="alert alert-danger hide" id="errorMsg" role="alert">Veuillez compléter tous
les champs obligatoires</div>
  <form role="form" method="POST" name="TitleForm">
    <label>Nom </label>
    <input class="form-control" type="text" name="lastName" placeholder="Nom"><br>
    <label>Email </label>
    <input class="form-control" data-mandatory="true" type="text" name="email"
placeholder="Email" required=""><br>
    <p class="invalid-feedback-email"></p>
    <label>Téléphone </label>
    <input class="form-control" type="text" name="mobile" placeholder="Téléphone"><br>
    <p class="invalid-feedback-mobile"></p>
    <label>Question ?</label>
    <input class="form-control" type="radio" name="field01" value="Choix 1">Choix 1
    <input class="form-control" type="radio" name="field01" value="Choix 2">Choix 2
    <input class="form-control" type="radio" name="field01" value="Choix 3">Choix 3
    <label>Select </label>
    <select class="form-control" name="field04">
      <option value="A">A</option>
      <option value="B">B</option>
      <option value="C">C</option></select>
    <button type="submit" name="formSubmit">Valider</button>
  </form>
</div>
```

Les input/select/textarea doivent contenir un attribut **name** qui contiendra le nom du champ de stockage parmi les valeurs suivantes correspondant à un contact de liste :

- civility
- firstName
- lastName
- email
- phone
- mobile
- fax
- address1
- address2
- zipcode
- city
- state
- country

- birthDate
- reference
- company
- field01
- field02
- field03
- field04
- field05
- field06
- field07
- field08
- field09
- field10
- field11
- field12
- field13
- field14
- field15

Les champs de type **text/date/textarea** doivent contenir l'attribut **placeholder="text"** qui servira de libellé.

Les champs de type **radio** doivent contenir l'attribut **value="value"** et **name** doit être aussi le champ field associé.

Les champs de type **select** doivent avoir des options pour définir les valeurs par défaut.

Pour la gestion des erreurs, on utilise 3 fonctionnalités propres aux formulaires :

- On place une div avec un **id="errorMsg"** en en-tête de formulaire, juste avant la balise <form>
- Pour afficher un message d'erreur après un email / mobile, on place une balise juste après l'input avec les classes **"invalid-feedback-email"** pour un email et **"invalid-feedback-mobile"** pour un champ mobile.
- Pour indiquer que le champ est obligatoire, on utilise **data-mandatory="true"**. On peut également utiliser **required="true"**

Astuce : Si le contact saisi via le formulaire existe déjà dans votre base de contact il sera alors mis à jour (il est identifié en suivant la règle de dédoublement du contrat).

Astuce : Le composant form ne doit être utilisé que dans une smartpage

Par ailleurs, il est aussi possible de configurer votre formulaire de saisie pour qu'il ne mette pas à jour votre base de contacts. En effet, en changeant la valeur de **name** des champs de formulaire en **ignoredField_XX** (XX étant une valeur numérique incrémentale : 1, 2, 3...), le champ ne sera pas pris en compte pour la création ou modification des contacts. Pour autant, **la réponse sera trouvable dans le fichier d'export des réponses** à votre formulaire.

De ce fait, si vous ignorez **tous les champs** appartenant à votre règle de dédoublement, **aucun** contact ne sera créé ni mis à jour et **aucune** liste de contact ne sera créée.

Exemple:

```
<div form-editor="">
  <h2>Title</h2>
```



```

    <h3>SubTitle</h3>
    <div class="alert alert-danger hide" id="errorMsg" role="alert">Veuillez compléter
tous les champs obligatoires</div>
    <form role="form" method="POST" name="TitleForm">
    <label>Nom </label>
    <input class="form-control" type="text" name="ignoredField_1" placeholder="Nom"><br>
    <label>Email </label>
    <input class="form-control" data-mandatory="true" type="text" name="ignoredField_2"
placeholder="Email" required=""><br>
    <p class="invalid-feedback-email"></p>
    <label>Téléphone </label>
    <input class="form-control" type="text" name="ignoredField_3"
placeholder="Téléphone"><br>
    <p class="invalid-feedback-mobile"></p>
    <label>Question ?</label>
    <input class="form-control" type="radio" name="ignoredField_4" value="Choix 1">Choix
1
    <input class="form-control" type="radio" name="ignoredField_4" value="Choix 2">Choix
2
    <input class="form-control" type="radio" name="ignoredField_4" value="Choix 3">Choix
3
    <label>Select </label>
    <select class="form-control" name="ignoredField_5">
        <option value="A">A</option>
        <option value="B">B</option>
        <option value="C">C</option></select>
    <button type="submit" name="formSubmit">Valider</button>
    </form>
</div>

```

1.2.5. Video

Il est possible d'ajouter un lecteur vidéo afin de diffuser une vidéo Youtube, Dailymotion ou Vimeo. Pour cela il faut utiliser une balise IMG avec l'attribut *video-editor*

Exemple de code

```

```

Astuce : Il est préférable de définir une source par défaut pour que l'aperçu avant édition soit correct (certains navigateurs n'affichent rien si un img n'a pas de source)

Astuce : Le composant video ne doit être utilisé que dans une smartpage

1.2.6. Social

Il est possible d'ajouter bloc contenant des liens sociaux vers votre smartpage. Pour cela il faut ajouter l'attribut *social-editor* sur une balise de type block (ex DIV, TD d'une table). Un lien par réseau sera ajouté via l'éditeur.

Le composant fera apparaître une zone avec des icônes en filigranes afin de vous permettre de cliquer dessus et de configurer le contenu social de votre création.

Exemple de code

```
<div social-editor=""></div>
```

1.2.7. Locator

Il est possible d'ajouter un bloc contenant le store locator vers votre smartpage. Pour cela il faut ajouter l'attribut *store-locator-editor* sur une balise de type block (ex DIV, TD d'une table) ainsi qu'un id='store-locator', contenant un input type text pour la recherche d'adresse (id='locator-geoloc-address'), un bouton type button (id='locator-search') et une balise bloc (ex div, td, etc.) pour la recherche geolocalisée (class="locator-geoloc") avec une ancre (a) (class="locator-geoloc-link").

En plus de cela, il est **obligatoire** d'ajouter un input de type hidden pour la geolocalisation (id='locator-geoloc-location') ainsi qu'une balise div pour les résultats (class='locator-results')

Exemple de code

```
<div store-locator-editor id="store-locator" align="center" class="wrapper pdt20">
  <div class="container">

<!-- facultatif -->
  <div class="dg-row" style="padding-bottom: 15px;">
    <div class="dg-col-xs-12 center store-locator-title">
      <div text-editor="">
        <p style="font-size: 18px; margin:0;">Rechercher un point de vente</p>
      </div>
    </div>
  </div>
<!-- fin du facultatif -->

  <div class="dg-row" style="padding-bottom: 15px;">
    <div class="dg-col-xs-12 dg-col-sm-8 dg-col-sm-offset-2 dg-col-md-6
dg-col-md-offset-3 center">
      <input type="hidden" id="locator-geoloc-location"/>
      <div class="input-group">
        <span class="input-group-btn">
          <button class="locator-geoloc-link btn btn-default">
            <i class="fa fa-crosshairs"></i>
```

```

        </button>
    </span>
    <input type="text" class="form-control" placeholder="Entrer une adresse"
id="locator-geoloc-address">
    <span class="input-group-btn">
        <button class="btn btn-default" id="locator-search"
type="button">Chercher</button>
    </span>
</div>
</div>
</div>
<div class="locator-results"></div>
</div>
</div>

```

1.3. Blocs déplaçable et supprimable

Il est possible de définir des blocs représentant une ligne qui agrège du contenu déplaçable ou supprimable. Chacun de ces blocs possède une "Drop-Zone" au-dessus et au-dessous permettant de déposer un autre bloc. Par défaut une "Drop-Zone" est présente (automatiquement ajouté par le système) en haut de création HTML.

Pour activer cette fonctionnalité il faut ajouter l'attribut *editable-block* sur une balise DIV. Toutes vos divs avec cet attribut doivent être au même niveau, et non contenus dans des sous divs.

Exemple de code

```

<div editable-block=''
  data-eb-name='le nom ou la description de mon bloc'
  data-eb-classes='[["dgf-edit"], ["dgf-circle"]]' >
  <div text-editor="">
    <p>YOUR TEXT</p>
  </div>
  
</div>

```

Vous pouvez verrouiller l'édition d'un block dans un modèle en ajoutant l'attribut "**uneditable**". Ce verrou pourra être activé/désactivé via l'éditeur avec un rôle gestionnaire de réseaux.

Astuce : les icônes doivent impérativement être entourées de double quote et non de simple quote car le flux de donnée est de type JSON.

2. Contraintes à respecter

2.1. Entête HTML

- Les créations HTML ne doivent en aucun cas utiliser du JS.
- Pas de lien vers des feuilles de style externes pour les créations pour le canal email, il faut embarquer le style directement dans une balise STYLE ou en inline sur les balises HTML

2.2. Lien de désabonnement

Pour tout email envoyé, nous devons fournir un moyen au destinataire de pouvoir se désinscrire de la liste d'envoi. Pour cela nous ajoutons automatiquement un lien en bas de page permettant de se désabonner.

Exemple de code

```
Click here to unsubscribe
```

Vous devez personnaliser ce lien en ajoutant simplement un lien pointant vers #OPTOUTLINK#

Exemple de code

```
<a href="#OPTOUTLINK#">Click here to unsubscribe</a>
```

2.3. Lien de preview

Pour tout email envoyé, nous pouvons fournir un moyen au destinataire de visualiser le mail dans son navigateur (sous-entendu en dehors du client mail).

Vous devez personnaliser ce lien en ajoutant simplement un lien pointant vers #PREVIEWLINK#

Exemple de code

```
<a href="#PREVIEWLINK#">Your preview message</a>
```

3. Astuces résolution incohérences d’affichages sur Outlook

Le logiciel Outlook utilise le moteur de rendu de word pour générer l’affichage des emails reçu. Ainsi on peut voir des différences plus ou moins importantes en fonction de la construction de l’email.

Sachez qu’il existe des outils pour tester l’affichage de vos emails sur toutes les versions de toutes les messageries. Par exemple : <https://www.emailonacid.com/> (solution payante).

3.1. Propriétés supplémentaires Outlook

Outlook ne prend pas toujours en compte certaines propriétés CSS, notamment de mise en page. Si les propriétés cellpadding et cellspacing ne sont pas définies à zéro, il est conseillé de l’ajouter dans votre code HTML.

Dans le cas ou certaines tables semblent décalées, c’est parce que par défaut Outlook ajoute probablement un padding. Pour corriger cela il faut ajouter les propriétés “mso-cellspacing: 0px; mso-padding-alt: 0px;” sur la <table>.

```
<table width="600" align="center" cellpadding="0" cellspacing="0" border="0" style="mso-cellspacing: 0px; mso-padding-alt: 0px; width:600px;">
  <tr>
    <td>
```

Vous pouvez ajouter un commentaire HTML spécifique pour Outlook (qui ne sera pas pris en compte par les autres boîtes de réception)

```
<!--[if (gte mso 9)|(IE)]>
<p> Mon contenu HTML uniquement spécifique pour Outlook & Internet Explorer </p>
<![endif]-->
```

Certaines versions d’Outlook ne savent pas interpréter pas les balises <div>. Vous pouvez remédier à cela en ajoutant une table autour de vos <div>, uniquement visible sur Outlook.

```
<!--[if (gte mso 9)|(IE)]>
<table width="100%" cellpadding="0" cellspacing="0" border="0" style="mso-cellspacing: 0px; mso-padding-alt: 0px;">
  <tr>
    <td width="285" valign="middle">
      <![endif]-->
      <div class="column" style="width: 100%; max-width: 285px; display: inline-block;">
        <table>
          <tr>
            <td>
              <![endif]-->
            </td>
          </tr>
        </table>
      </div>
      <!--[IF (gte mso 9)|(IE)]>
    </td>
  </tr>
</table>
<![endif]-->
```

3.2. Corrections d'affichage

3.2.1. Règles générales

Sur Outlook, le padding sur les paragraphes n'est pas supporté. Si un padding est appliqué sur un paragraphe, Outlook interprète le padding d'un <p> en margin. Ce margin prendra le dessus sur l'éventuel margin que vous aurez préalablement appliqué.

Pour éviter cela, nous vous conseillons de gérer les espaces en mettant des paddings sur des balises <td>, qui ne seront pas modifiés. Vous assurez ainsi l'affichage.

3.2.2. Espaces supplémentaires

Dans le cas où des espaces supplémentaires sont affichés lors de la réception de votre email sur Outlook :

- il faut ajouter les propriétés display:block; et margin:0; sur toutes vos balises <p> et .
- il faut définir border-spacing: 0; ainsi que border-collapse: collapse; sur vos <table> et <td> pour supprimer des petits espacements non souhaités.

3.2.3. Images

Vos images doivent faire la taille exacte affichée sur votre email. Dans le cas où celle-ci est trop grande, tout l'affichage du mail sera décalé vers la droite.

Si malgré cela, vos images ne s'affichent pas correctement, vous pouvez fixer la taille de vos images en spécifiant des valeurs width et height en CSS, mais aussi avec width="x" et height="y".

3.2.4. Liens

Si vous avez défini une couleur de texte sur votre lien et qu'elle est remplacée par la couleur bleue sur Outlook, ajoutez une balise avec vos propriétés CSS en inline à l'intérieur de votre balise <a>

```
<a href="#" style="color:#FF0000;"><span style="color:#FF0000;"> Texte de lien </span></a>
```

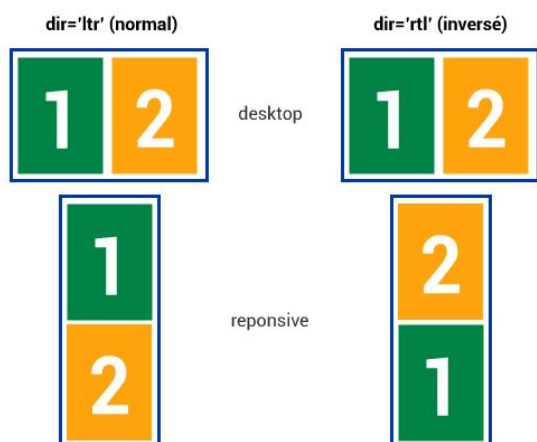
3.2.5. Texte de droite à gauche

Sur Outlook, un texte écrit de droite à gauche est dû à une base dir="rtl". Outlook ne gère pas bien cette propriété, mais il est possible de remettre le texte dans le bon sens, tout en gardant dir="rtl".

Pour cela il suffit de mettre la propriété dir="ltr" sur la balise <p> ou le texte est inversé

```
<td dir="rtl" style="border-spacing: 0px; border-collapse: collapse; padding: 0;">
  <a dir="ltr" href="#" target="_blank">
    Texte de lien
  </a>
  <p dir="ltr" style="display: block; border-collapse: collapse; margin: 0;">
    Lorem ipsum dolor sit amet, consectetur adipiscing eliada a enim vel.
  </p>
</td>
```

Fonctionnement de dir="rtl"



Copyright

L'ensemble de ce code relève de la législation française et internationale sur le droit d'auteur et la propriété intellectuelle. Tous les droits de reproduction sont réservés, y compris pour les documents téléchargeables et les représentations iconographiques et photographiques. La reproduction de tout ou partie de ce code sur quelque support que ce soit est formellement interdite sauf autorisation écrite émanant de la société Digitaleo.