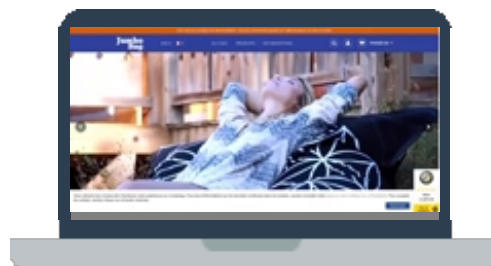




Rapport d'analyse performance et qualité



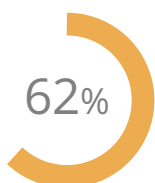
<https://dev.jumbobag.fr>

Ce rapport a été généré sur [Dareboost.com](https://dareboost.com), service en ligne d'analyse et de surveillance de la performance et de la qualité des sites web.

Sommaire

Résumé	3
Conseils et bonnes pratiques	4
Accessibilité	4
Nombre de requêtes	5
Optimisation du rendu	7
Politique de cache	9
Qualité	11
Référencement (SEO)	15
Sécurité	18
Volume de données	22
Apache	26
jQuery	27

Résumé



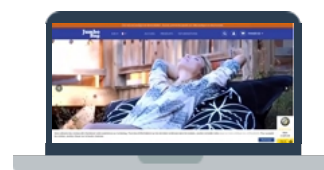
Problèmes



Améliorations



Succès



VISITEUR SIMULÉ :



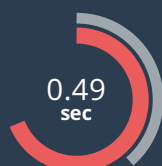
Chrome



Paris

8.0/1.5Mbps (Latence : 50 ms)

Premier octet



Début affichage



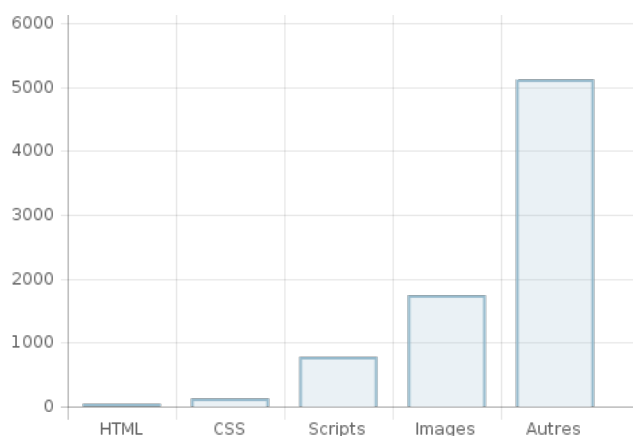
Fin chargement



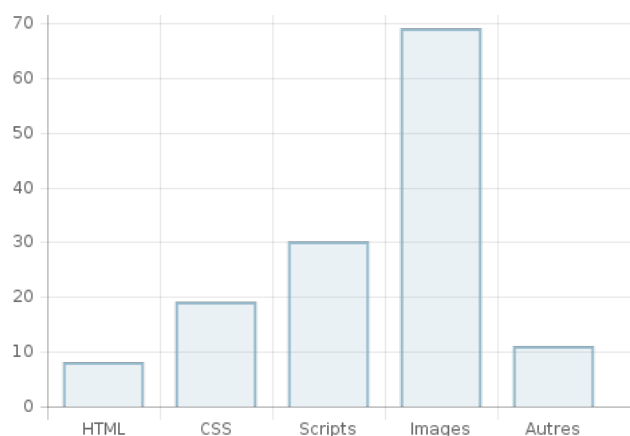
Speed Index



Répartition du poids



Répartition du nombre de requêtes



Technologies détectées



Apache



Font Awesome



Google Analytics



Google Font API



Google Tag Manager



Magento



Prototype



Twitter Bootstrap



jQuery



PHP

Conseils et bonnes pratiques

Accessibilité

Vos principaux axes d'optimisation

0/100

#2381

! Expliquez le but de chaque champ de formulaire

Améliorez l'expérience utilisateur sur votre site en spécifiant une étiquette descriptive pour chaque champ de formulaire.

Un formulaire est composé de plusieurs champs qui doivent être les plus explicites possibles pour que l'utilisateur comprenne rapidement leur fonction.

Définir un label

Pour indiquer l'objectif d'un champ, nous vous recommandons l'utilisation de la balise `label` :

```
<label for="nom">Renseignez votre nom :</label>
<input id="nom" type="text" name="nom">
```

Exemple

Si vous ne souhaitez pas afficher de label, vous pouvez utiliser les attributs `aria-label` ou `title` (ce dernier n'est cependant pas supporté par certaines liseuses d'écran).

En savoir plus sur [l'association de texte à un champ de formulaire](#).

Cette page contient **3 champs sans explications sur leur objectif** :

- ```
<input id="search" type="text" name="q"
value="" class="input-text"/>
```
- ```
<select name="currency_switch" onchange="window.location=this.value">
```
- ```
<select name="language_switch" onchange="window.location=this.value">
```

0/100

#2524

#### ! 1 bouton est vide

L'utilisateur doit être informé de l'action qui sera déclenchée s'il clique sur un bouton.

L'élément `<button>` ne peut être vide. Vous devez y indiquer du texte, ou encore une image avec un texte alternatif pour permettre à tous vos internautes d'être informés des conséquences de l'action.

Exemple :

```
<button>
 Télécharger
</button>
```

### Vos principaux axes d'optimisation

6/100

#2442

#### ! Économisez 17 requêtes à l'aide de sprites CSS

Combiner les petites images dans une sprite CSS réduit le nombre de fichiers que le navigateur doit télécharger et accélère donc le chargement de votre page.

##### Les sprites CSS

Une sprite CSS est un fichier unique dans lequel plusieurs petites images sont regroupées, positionnées les unes à côté des autres. L'affichage de chaque petite image au sein de votre page se fait ensuite par l'application de styles CSS. Une seule requête est alors nécessaire pour récupérer l'ensemble des images. Cette technique est à utiliser uniquement pour les petites images, comme des icônes par exemple, afin que la sprite CSS ne soit pas trop lourde.

##### Exemple

Voici un exemple de sprite CSS, regroupant plusieurs icônes :



La page applique par ailleurs les styles associés à la sprite, du type :

```
.sprite {
 background-image: url(img/sprite.png);
 background-repeat: no-repeat;
 display: block;
}

.sprite-browsers-firefox {
 width: 31px;
 height: 28px;
 background-position: -74px 0;
}
```

Exemple

Il reste à définir la bonne classe dans votre fichier HTML, et l'icône apparaît :

```

```

Exemple

##### Comment créer une sprite CSS ?

Des outils facilitent leur création qui peut se révéler complexe. Par exemple :

- <https://draeton.github.io/stitches/>
- <http://spritegen.website-performance.org/>

Vous obtenez alors votre sprite CSS avec les styles à appliquer.

**18 images peuvent être regroupées au sein d'une sprite CSS.** Le nom de domaine jumbobag devrait utiliser cette technique pour les ressources suivantes :

- <https://dev.jumbobag.fr/skin/frontend/jumbobag/default/images/twitter-footer.png>
- [dev.jumbobag.fr/media/wysiwyg/1416523839\\_payment\[...\]method\\_card\\_visa-128.png](https://dev.jumbobag.fr/media/wysiwyg/1416523839_payment[...]method_card_visa-128.png)
- [dev.jumbobag.fr/skin/frontend/jumbobag/default/i\[...\]ginal-bl-price-image.png](https://dev.jumbobag.fr/skin/frontend/jumbobag/default/i[...]ginal-bl-price-image.png)
- <https://dev.jumbobag.fr/skin/frontend/jumbobag/default/images/prev-button.png>
- [dev.jumbobag.fr/skin/frontend/jumbobag/default/i\[...\]ages/facebook-footer.png](https://dev.jumbobag.fr/skin/frontend/jumbobag/default/i[...]ages/facebook-footer.png)
- <https://dev.jumbobag.fr/skin/frontend/jumbobag/default/images/next-button.png>
- [dev.jumbobag.fr/media/wysiwyg/1416523842\\_payment\[...\]thod\\_master\\_card-128.png](https://dev.jumbobag.fr/media/wysiwyg/1416523842_payment[...]thod_master_card-128.png)
- <https://dev.jumbobag.fr/skin/frontend/jumbobag/default/images/picot-bleu.png>
- [dev.jumbobag.fr/skin/frontend/jumbobag/default/j\[...\]ybox/fancybox\\_sprite.png](https://dev.jumbobag.fr/skin/frontend/jumbobag/default/j[...]ybox/fancybox_sprite.png)
- <https://dev.jumbobag.fr/skin/frontend/jumbobag/default/images/mail-footer.png>
- et 8 autres

## 33/100

#2418

### ! Regroupez 18 fichiers JavaScript

Chaque requête HTTP représente un coût en terme de performance (roundtrip time, utilisation de bande passante...).

Il est ainsi préférable de faire une requête vers un fichier de 50ko de données plutôt que 10 requêtes vers des fichiers de 5ko.

#### Quelle répartition adopter ?

Répartissez vos scripts en les intégrant directement dans votre code HTML ou en les regroupant dans des fichiers de taille plus conséquente. Nous vous conseillons l'emploi de cette dernière méthode pour profiter des mécanismes de cache.

Vous devriez reconsidérer les ressources suivantes :

- [dev.jumbobag.fr/js/queldorei/jquery.the\[...\].nch.plugins.min.js](#)
- [dev.jumbobag.fr/js/queldorei/jquery.the\[...\].revolution.min.js](#)
- [dev.jumbobag.fr/skin/adminhtml/base/def\[...\].jquery-1.12.0.min.js](#)
- [dev.jumbobag.fr/skin/adminhtml/base/def\[...\].jopupBuilder/jsg.js](#)
- et 14 autres

## 41/100

#2419

### ! Regroupez 12 fichiers CSS

Chaque requête HTTP représente un coût en terme de performance (roundtrip time, utilisation de bande passante...).

Il est ainsi préférable de faire une requête vers un fichier de 50ko de données plutôt que 10 requêtes vers des fichiers de 5ko.

#### Quelle répartition adopter ?

Répartissez vos styles en les intégrant directement dans votre code HTML ou en les regroupant dans des fichiers de taille plus conséquente. Nous vous conseillons l'emploi de cette dernière méthode pour profiter des mécanismes de cache.

Vous devriez reconsidérer les ressources suivantes :

- [dev.jumbobag.fr/skin/adminhtml/base/def\[...\].rbox/colorbox5.css](#)
- [dev.jumbobag.fr/skin/adminhtml/base/def\[...\].rbox/colorbox3.css](#)
- [dev.jumbobag.fr/skin/frontend/jumbobag/\[...\].jasonry-gallery.css](#)
- [dev.jumbobag.fr/skin/frontend/jumbobag/\[...\]lt/css/style-r.css](#)
- et 8 autres

### Vos principaux axes d'optimisation

0/100

#2531

#### ! 1 dépendance critique détectée

La défaillance d'un fournisseur de contenu externe peut rendre indisponible votre site.

##### Les Single Point Of Failure

Un Single Point Of Failure (SPOF) front-end se caractérise par une dépendance critique d'une page web à un contenu externe, susceptible d'empêcher totalement l'affichage de la page en cas de problème avec le fournisseur externe.

Par exemple, si votre page web appelle un script bloquant hébergé sur les serveurs de Google, votre page se trouve dépendante de toute défaillance provenant de l'appel à ce script. Voir [cet article sur les SPOF](#) pour en savoir plus.

##### Que faire pour éviter les SPOF ?

**Éliminez ces dépendances autant que possible, même si elles concernent des fournisseurs renommés.** Si vous devez faire appel à un contenu externe, assurez-vous que l'appel est effectué de manière asynchrone, et que vous disposez de fallbacks (alternatives en cas d'échec), afin de maîtriser le comportement de la page en cas de panne d'un service externe.

Nous nous assurons ici que la page analysée ne dépend pas de manière critique de ressources externes parmi les plus connues (googleapis, typekit, etc). Il s'agit en effet de cas de Single Point Of Failure front-end

La ressource suivante constitue un SPOF pour cette page :

- [//fonts.googleapis.com/css?family=Open+Sans:300,400,700](https://fonts.googleapis.com/css?family=Open+Sans:300,400,700)

0/100

#2353

#### ! Différez l'utilisation du code JavaScript

Lorsque le navigateur web rencontre du code JavaScript en interprétant le code source d'une page web, cela peut ralentir considérablement l'affichage de la page, surtout s'il est nécessaire de télécharger un script externe.

Différez au maximum l'utilisation du JavaScript pour assurer un début rapide de l'affichage de la page.

##### Que faire ?

Utilisez si possible l'une des techniques suivantes pour des appels à des fichiers externes :

- utilisation de l'attribut [async](#)
- utilisation de l'attribut [defer](#)
- ajout du script dans le DOM en JavaScript, lors de l'évènement onload
- placer les scripts à la fin de votre code source (idéalement à la fin du <body>)

916.8 Ko du code JavaScript sont analysés lors du chargement initial de la page. Différez l'analyse de ce code pour éviter de bloquer l'affichage de la page.

- [dev.jumbobag.fr/me\[...\].js](#) (716.7 Ko)
- [dev.jumbobag.fr/sk\[...\].n.js](#) (87.3 Ko)
- [dev.jumbobag.fr/js\[...\].n.js](#) (42.5 Ko)
- [dev.jumbobag.fr/sk\[...\].n.js](#) (17.3 Ko)
- [dev.jumbobag.fr/js\[...\].n.js](#) (12.2 Ko)
- [https://dev.jumbobag.fr/](#) (9.9 Ko de code JavaScript intégré)
- [dev.jumbobag.fr/sk\[...\].d.js](#) (9.7 Ko)
- [dev.jumbobag.fr/sk\[...\].n.js](#) (6.4 Ko)
- [dev.jumbobag.fr/sk\[...\].n.js](#) (5.5 Ko)
- [dev.jumbobag.fr/sk\[...\].t.js](#) (2.4 Ko)
- [dev.jumbobag.fr/sk\[...\].o.js](#) (2.2 Ko)
- [dev.jumbobag.fr/sk\[...\].t.js](#) (1.8 Ko)
- [dev.jumbobag.fr/sk\[...\].e.js](#) (1.7 Ko)
- [dev.jumbobag.fr/sk\[...\].e.js](#) (1.2 Ko)

0/100

#2550

### ! 2 scripts peuvent être injectés de façon plus optimale

L'injection de scripts via l'instruction JavaScript `document.write` retarde l'affichage d'une partie au moins de votre page et/ou l'interactivité de votre page pour l'utilisateur.

#### Besoin d'injecter un script ?

`document.write` est parfois utilisé à tort pour injecter un script. Exemple :

```
document.write('<script src="" + src + "" type="text/javascript"></script>');
```

Exemple

Comme indiqué sur cet article (EN), vous devriez plutôt envisager l'utilisation du pattern "createElement-insertBefore" :

```
var sNew = document.createElement("script");
sNew.async = true;
sNew.src = "http://ajax.googleapis.com/ajax/libs/jquery/1.5.1/jquery.min.js";
var s0 = document.getElementsByTagName('script')[0];
s0.parentNode.insertBefore(sNew, s0);
```

Exemple

Sur cette page, 2 scripts sont injectés suite à l'utilisation de l'instruction `document.write` :

Scripts injectés par <https://dev.jumbobag.fr/> (inline 2)

- `document.write('<script type="text/javascript" src="//ajax.googleapis.com/ajax/libs/jquery/1.8.0/jquery.min.js">')`

Scripts injectés par [static-eu.payments-amazon.com/OffAmazonPay\[...\]de/lpa/js/Widgets.js](https://static-eu.payments-amazon.com/OffAmazonPay[...]de/lpa/js/Widgets.js)

- `document.write("<script type='text/javascript' src='"+OffAmazonPayments.Loader.loginJsUrl+" id= '"+OffAmazonPayments.Loader.LOGINJS_NAME+"></script>")`

## Les autres conseils

89/100

#2416

### ! Vous devriez réduire le nombre d'éléments du DOM

Une structure complexe de la page implique plus d'octets à télécharger, et complexifie la recherche d'éléments précis.

Une page bien conçue peut présenter des contenus riches tout en conservant un nombre d'éléments raisonnable. [En savoir plus \(EN\)](#).

Nous vous conseillons de conserver une page avec moins de 1000 éléments dans le DOM.

Le DOM de cette page contient trop d'éléments (1092 éléments).



### Vos principaux axes d'optimisation

0/100

#70

#### ! N'utilisez pas de scripts inline trop longs

Vous devriez permettre la mise en cache de tout script ayant une taille non négligeable, afin de réduire le temps de chargement de vos pages.

##### Scripts inline / mise en cache

Les scripts dits "inline" permettent d'intégrer facilement de petites portions de scripts directement dans le code HTML. Exemple :

```
<script type="text/javascript">
 (function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=...'/analytics.js','ga');
 ga('create', 'UA-11111111-1', 'monsite.com');
</script>
```

Exemple

En procédant ainsi, vous évitez d'effectuer une requête vers le serveur pour récupérer la ressource. Les scripts inline représentent alors un gain de performance pour intégrer de petits scripts.

Dès lors qu'un script dispose d'une taille assez conséquente, nous vous conseillons de l'externaliser et d'effectuer une requête pour le récupérer. Cela vous permettra de bénéficier des mécanismes de cache sur la ressource.

##### Que faire ?

Externalisez vos scripts de plus de 1 500 caractères dans un ou des fichiers distincts.

Vous devriez **regrouper les scripts suivants dans un/des fichiers distincts** :

- ```
//<![CDATA[
  var Translator = new Translate({"HTML tags are not allowed":"Les balises HTML ne sont pas autoris\u00e9es","Please select an optio...
```
- ```
SG_POPUP_DATA[6] = {"id":"6","title":"exit","type":"exitintent","effect":"pulse","0":"width","1":"","height":"","delay":10,"duration":1,"2":"initialWid...
```
- ```
(function(jQuery) {

  /*****
  - PREPARE PLACEHOLDER FOR SLIDER -
  *****/

  var htmlDivCss = unescape(".hesperiden.tparrows%20%7B%0A%09cursor%3Apointer%3B%0A%09background%3Argb%280%2C0%2C0%2C0.5%29%3B%0A%09width%3A40px%3B%0A%...
```
- ```
(function(jQuery) {

 /*****
 - PREPARE PLACEHOLDER FOR SLIDER -
 *****/

 var htmlDivCss = unescape(".hesperiden.tparrows%20%7B%0A%09cursor%3Apointer%3B%0A%09background%3Argb%280%2C0%2C0%2C0.5%29%3B%0A%09width%3A40px%3B%0A%...
```

## ✓ 8 requêtes provenant de partis tiers n'adoptent pas une politique de cache assez longue

En permettant la conservation de vos ressources statiques plusieurs jours en cache, vous diminuez la charge de votre serveur.

### L'entête expires

Certaines de vos ressources utilisent l'en-tête HTTP **Expires** pour disposer d'une politique de cache efficace, ce qui correspond à une bonne pratique. Cependant, vous devriez envisager d'améliorer sa configuration pour profiter au mieux des mécanismes de cache. L'en-tête HTTP **Expires** se présente sous la forme suivante :

```
Expires: Thu, 25 Dec 2014 20:00:00 GMT
```

[Exemple](#)

Lorsque vous mettez en production **une nouvelle version de votre site, pensez bien à renommer les ressources statiques** ayant été modifiées (versioning). En effet, si vous ne changez pas leur nom, vos utilisateurs garderont les ressources stockées en cache, correspondant à l'ancienne version. Ils risquent alors de se retrouver dans une version instable de votre page. Par exemple :

```
maresource.min.20140101.js
```

[Exemple](#)

Voir [les recommandations de Yahoo!](#) sur le sujet.

### Quelle configuration adopter ?

Nous vous conseillons de paramétrer l'en-tête **Expires** afin que la date mentionnée soit comprise entre 2 jours et 1 an par rapport à aujourd'hui.

Cette page contient 8 ressources **n'ayant pas une date d'expiration assez lointaine** :

*Les ressources suivantes sont hébergées par un parti tiers, il se peut donc qu'elles ne soient pas sous votre responsabilité. Vous devriez cependant considérer toute alternative possible à ces fichiers pour rester en accord avec la bonne pratique.*

- [static-eu.payments-amazon.com/OffAmazon\[...\]/lpa/js/Widgets.js](#)
- [https://www.googletagmanager.com/gtm.js?id=GTM-T55T8X](#)
- [https://www.google-analytics.com/analytics.js](#)
- [widgets.trustedshops.com/js/X3B73FD19E1\[...\]/DD7DD1C76F880D7.js](#)
- [https://www.google-analytics.com/plugins/ua/ec.js](#)
- [https://eu.account.amazon.com/lwa/js/sdk/login1.js](#)
- [widgets.trustedshops.com/css/style.css\[...\]/elementId=&channel=](#)
- [widgets.trustedshops.com/css/tbAuthent\[...\]/F81DD7DD1C76F880D7](#)

## Vos principaux axes d'optimisation

0/100

#2491

**! Plusieurs éléments utilisent le même identifiant**

Utiliser un même identifiant pour 2 éléments différents peut entraîner des effets de bord, notamment lors de l'exécution de JavaScript ou lors de l'application des règles CSS.

**Identifiant d'éléments**

Chaque élément d'une page web peut disposer d'un identifiant, défini par l'attribut `id` :

```
<p>

</p>
```

Exemple

Ces identifiants vous permettent de manipuler les éléments de votre page avec des instructions CSS ou JavaScript.

**Que faire ?**

Vous devez faire en sorte qu'aucun élément de la page ne dispose du même identifiant. Si vous souhaitez partager des propriétés/comportements entre plusieurs éléments, il faudra utiliser l'attribut `class`, dédié à cet effet :

```
<p>

</p>
```

Exemple

Les identifiants suivants sont utilisés plusieurs fois dans la page :

- `gammes`, utilisé 3 fois
- `original`, utilisé 3 fois
- `printed`, utilisé 3 fois
- `x-trem`, utilisé 3 fois
- `flottant`, utilisé 3 fois

0/100

#2511

**! 75 sélecteurs CSS sont dupliqués**

Utiliser plusieurs fois le même sélecteur sur des blocs déclaratifs différents au sein d'un même fichier CSS peut nuire à la lisibilité et à la maintenabilité du code. C'est également une source d'optimisation, puisque regrouper ces éléments au sein d'une même règle réduit le poids du fichier.

**Les règles CSS**

Les règles CSS permettent de sélectionner un ensemble d'éléments du DOM (votre html), et de leur appliquer des propriétés de mise en forme.

**Comment améliorer ce point ?**

Il vous suffit de regrouper au sein d'une même règle les sélecteurs.  
Par exemple, si votre fichier dispose des règles suivantes :

```
.maClasse {
 margin: 0;
}
...
.maClasse{
 border: 1px solid black;
}
```

Vous devriez envisager leur réécriture de cette manière :

```
.maClasse {
 margin: 0;
 border: 1px solid black;
}
```

Si la duplication des selecteurs est volontaire (architecture propre a votre code, etc), vous pouvez envisager l'utilisation d'un outil de minification CSS capable de faire cette transformation automatiquement.

**3859 caractères peuvent être économisés.** Les fichiers suivants définissent des sélecteurs identiques dans des règles différentes :

[dev.jumbobag.fr/skin/adminhtml/base/default\[...\]lorbox/colorbox4.css](#)

- `#sgcboxOverlay.colorbox4` (2 fois)
- `.colorbox4 #sgcboxContent` (2 fois)
- `.colorbox4 #sgcboxLoadedContent` (2 fois)
- `.colorbox4 #sgcboxTitle` (2 fois)
- `.colorbox4 .sgcboxIframe` (2 fois)

<https://dev.jumbobag.fr/options.css.php?store=default>

- `.slider-container` (2 fois)
- `.footer-info` (2 fois)

[https://dev.jumbobag.fr/skin/frontend/jumbobag/default/css/style-r.css](#)

- `.stars-p` (10 fois)
- `.catalog-product-view .i[...]ntainer .swatch-img:not(:last-child)` (10 fois)
- `.product-view .add-to-cart button.button span span` (10 fois)
- `@media (max-width: 480px) .uvmg1` (10 fois)
- `@media (max-width: 480px) .thin-part` (10 fois)
- `@media (min-width: 1024px) .uvmg1` (10 fois)
- `.marg-plus-tissu` (10 fois)
- `.product-original-printe[...]jissu .container .wheelmark h3:before` (2 fois)
- et 9 autres

[dev.jumbobag.fr/skin/adminhtml/base/default\[...\]lorbox/colorbox2.css](#)

- `#sgcboxOverlay.colorbox2` (2 fois)
- `.colorbox2 #sgcboxContent` (2 fois)
- `.colorbox2 #sgcboxLoadedContent` (2 fois)
- `.colorbox2 #sgcboxTitle` (2 fois)
- `.colorbox2 .sgcboxIframe` (2 fois)

[dev.jumbobag.fr/skin/adminhtml/base/default\[...\]lorbox/colorbox1.css](#)

- `#sgcboxOverlay.colorbox1` (2 fois)
- `.colorbox1 #sgcboxContent` (2 fois)
- `.colorbox1 #sgcboxLoadedContent` (2 fois)
- `.colorbox1 .sgcboxIframe` (2 fois)

[dev.jumbobag.fr/media/css\\_secure/881e23f6d\[...\]b39bbec31a991d4a.css](#)

- `.toolbar` (3 fois)
- `footer .liens-footer ul` (3 fois)
- `footer .liens-footer` (2 fois)
- `.form-list .customer-name-prefix-suffix .name-lastname` (2 fois)
- `.form-list .customer-nam[...]ffix .name-lastname input.input-text` (2 fois)
- `.breadcrumbs` (2 fois)
- `@media (min-width: 768px) .col-sm-3` (2 fois)
- `.toolbar .pager` (2 fois)
- et 25 autres

[dev.jumbobag.fr/skin/adminhtml/base/default\[...\]lorbox/colorbox5.css](#)

- `#sgcboxOverlay.colorbox5` (2 fois)
- `.colorbox5 #sgcboxContent` (2 fois)
- `.colorbox5 #sgcboxLoadedContent` (2 fois)
- `.colorbox5 #sgcboxTitle` (2 fois)
- `.colorbox5 .sgcboxIframe` (2 fois)

[dev.jumbobag.fr/skin/adminhtml/base/default\[...\]lorbox/colorbox3.css](#)

- `#sgcboxOverlay.colorbox3` (2 fois)
- `.colorbox3 #sgcboxContent` (2 fois)
- `.colorbox3 #sgcboxLoadedContent` (2 fois)
- `.colorbox3 #sgcboxTitle` (2 fois)

## ! Le mot clé !important est utilisé 291 fois

Le mot clé !important s'approche d'un hack permettant d'interdire toute surcharge d'une propriété définie. Si parfois il peut s'avérer utile, cela doit rester exceptionnel !

Si vous en abusez, songez à revoir votre structure de code CSS. Dans ce conseil, vous êtes pénalisé au delà de 10 utilisations de ce mot clef.

Voici la liste des !important détectés :

<https://dev.jumbobag.fr/options.css.php?store=default>

- `.fullwidthbanner-container .fullwidthbanner {max-height: 460...` (ligne 187, col 47)

<https://dev.jumbobag.fr/skin/frontend/jumbobag/default/css/style-r.css>

- `.product-original .presentation-produit-wrapper {background:...` (ligne 5, col 2)
- `.product-original .entretien-wrapper {background: transparen...` (ligne 9, col 2)
- `.product-original .wrapper-en-savoir-plus .inner-en-savoir-p...` (ligne 13, col 2)
- `.product-original .informations-produit .product-infos .prod...` (ligne 27, col 2)
- `.product-original .height-centered h1 .title-product-h1 {dis...` (ligne 34, col 2)
- `.product-original .informations-produit .product-images {mar...` (ligne 44, col 2)
- `.product-original .presentation-produit {padding-bottom: 110...` (ligne 53, col 2)
- `.product-original .informations-produit .product-infos .prod...` (ligne 69, col 2)
- et 160 autres

[dev.jumbobag.fr/skin/adminhtml/base/default\[...\]/pBuilder/animate.css](https://dev.jumbobag.fr/skin/adminhtml/base/default[...]/pBuilder/animate.css)

- `.flipInX {backface-visibility: visible !important}` (ligne 1530, col 5)
- `.flipInX, .flipInY {-webkit-backface-visibility: visible !imp...` (ligne 1536, col 5)
- `.flipInY {backface-visibility: visible !important}` (ligne 1595, col 5)
- `.flipOutX {backface-visibility: visible !important}` (ligne 1634, col 5)
- `.flipOutX, .flipOutY {-webkit-backface-visibility: visible !i...` (ligne 1638, col 5)
- `.flipOutY {backface-visibility: visible !important}` (ligne 1673, col 5)

[dev.jumbobag.fr/skin/frontend/jumbobag/def\[...\]/css/animatehome.css](https://dev.jumbobag.fr/skin/frontend/jumbobag/def[...]/css/animatehome.css)

- `.flipInX {-webkit-backface-visibility: visible !important}` (ligne 1980, col 3)
- `.flipInX {backface-visibility: visible !important}` (ligne 1981, col 3)
- `.flipInY {-webkit-backface-visibility: visible !important}` (ligne 2053, col 3)
- `.flipInY {backface-visibility: visible !important}` (ligne 2054, col 3)
- `.flipOutX {-webkit-backface-visibility: visible !important}` (ligne 2100, col 3)
- `.flipOutX {backface-visibility: visible !important}` (ligne 2101, col 3)
- `.flipOutY {-webkit-backface-visibility: visible !important}` (ligne 2143, col 3)
- `.flipOutY {backface-visibility: visible !important}` (ligne 2144, col 3)

[dev.jumbobag.fr/media/css\\_secure/881e23f6d\[...\]/b39bbec31a991d4a.css](https://dev.jumbobag.fr/media/css_secure/881e23f6d[...]/b39bbec31a991d4a.css)

- `h1 {letter-spacing: -1px !important}` (ligne 22, col 55)
- `h2 {letter-spacing: -1px !important}` (ligne 23, col 55)
- `h3 {letter-spacing: -1px !important}` (ligne 24, col 53)
- `h4 {letter-spacing: 1px !important}` (ligne 25, col 36)
- `h5 {letter-spacing: -1px !important}` (ligne 26, col 36)
- `h6 {letter-spacing: -1px !important}` (ligne 27, col 36)
- `.hidden {display: block !important}` (ligne 43, col 9)
- `.hidden {border: 0 !important}` (ligne 43, col 34)
- et 100 autres

## Les autres conseils

## ! Séparez vos styles de votre code HTML

Dissocier les balises HTML de leur style améliore la maintenabilité du code et favorise sa factorisation.

### Définir les styles CSS

Les styles CSS permettent de mettre en forme la page. Pour les définir, vous disposez de 3 méthodes principales :

- la déclaration dans un fichier CSS
- la déclaration dans un bloc "inline" (balises <style>)
- la déclaration dans un attribut style d'un élément HTML

### Que faire ?

Nous vous conseillons de regrouper vos styles CSS au sein de balises `<style>` ou dans des fichiers CSS dédiés. Ainsi, la partie HTML se charge uniquement de fournir la structure de la page, et sa mise en forme est externalisée. L'attribut `style` devrait uniquement être généré en cas de besoin par du code JavaScript (exemple : nécessité de connaître la taille de l'écran).

Cette page utilise 9 attribut(s) `style` :

- `<a style="background-image:url('https://dev.jumbobag.fr/skin/frontend/jumbobag/default/images/flags/default.png');" href="https://dev.jumbobag.fr/?__store=default&__from_store=default">`
- `<a style="background-image:url('https://dev.jumbobag.fr/skin/frontend/jumbobag/default/images/flags/en.png');" href="https://dev.jumbobag.fr/?__store=en&__from_store=default">`
- `<a style="background-image:url('https://dev.jumbobag.fr/skin/frontend/jumbobag/default/images/flags/it.png');" href="https://dev.jumbobag.fr/?__store=it&__from_store=default">`
- `<a style="background-image:url('https://dev.jumbobag.fr/skin/frontend/jumbobag/default/images/flags/de.png');" href="https://dev.jumbobag.fr/?__store=de&__from_store=default">`
- et 5 autres

### Vos principaux axes d'optimisation

0/100

#69

#### ! Ajoutez l'attribut alt sur vos balises <img>

L'attribut `alt` est un critère important en terme de référencement (SEO). En effet, les robots d'indexation des moteurs de recherche ne peuvent pas analyser de contenu graphique. Il se servent alors de leur texte alternatif pour répondre aux requêtes des internautes. [C'est par exemple le cas pour Google images.](#)

```

```

Exemple

De plus, l'attribut `alt` s'avère utile dans d'autres cas :

- utilisation d'un lecteur d'écran (personnes malvoyantes)
- connexion trop lentes pour charger l'image
- affichage d'un contenu dans le cas d'une erreur dans l'attribut `src`

Vous avez 64 balises `img`, mais 23 d'entre elles ne définissent pas d'attribut `alt` :

- ``
- ``
- ``
- ``
- ``
- ``
- ``
- ``
- ``
- ``
- et 13 autres

**Attention**, notez cependant que 5 de vos images contiennent un attribut `alt`, **mais ils sont vides** :

- ``
- ``
- ``

Il reste possible de définir un texte alternatif vide si aucune description ne semble cohérente pour l'image, mais soyez vigilant. Il est préférable de veiller à ce que la majorité de vos images indiquent un texte cohérent. Voir [les recommandations du W3C \(EN\)](#).

0/100

#2475

### ! Indiquez la présence d'au moins un produit sur la page

Les moteurs de recherche se servent des données structurées dans le but de mettre en valeur les informations utiles de la page.

#### Les données structurées

Vous pouvez ajouter des informations sur le sens de certains éléments de votre page (données sémantiques). Ainsi, vous pouvez indiquer que telle section parle d'un produit, une autre parle d'une personne, d'une entreprise, etc. Un moteur de recherche pourra alors extraire ces informations et les mettre en avant sur ses pages de résultats (par exemple la note attribuée à un produit, son prix, ...).

#### Comment les définir ?

Ces données sémantiques peuvent être définies au choix de 3 façons différentes : les microdonnées, JSON-LD, et RDFa (l'utilisation des 2 premières reste préférée pour leur simplicité).

Voici un exemple d'utilisation des microdonnées pour spécifier la présence d'un produit :

```
<div itemscope="" itemtype="http://schema.org/Product">

 BRAND

 <div itemprop="offers" itemscope itemtype="http://schema.org/Offer">
 4399 p.
 </div>
</div>
```

Exemple

Vous pouvez consulter [la liste des propriétés fournies par Schema.org](#). Une page ecommerce peut par exemple se servir des propriétés **Product**, **Offer**, **Order**, ou encore **Rating**.

Nous vous conseillons d'utiliser des données structurées liées au ecommerce sur cette page (toutes les pages de votre site doivent notamment faire référence à un produit).

0/100

#2399

### ! Améliorez la lecture de votre site par les réseaux sociaux

Transmettre aux réseaux sociaux des informations liées à la page vous permet de renforcer votre référencement.

#### Les propriétés Open Graph

Plusieurs propriétés permettent aux réseaux sociaux d'en savoir plus sur le contenu de la page. Nous vous recommandons d'utiliser au moins les quatre propriétés suivantes :

- `<meta property="og:title" content="The title" />`
- `<meta property="og:type" content="The type" />`
- `<meta property="og:url" content="http://url.com/" />`
- `<meta property="og:image" content="http://image.jpg" />`

Exemple

Exemple

Exemple

Exemple

Ces informations permettent d'améliorer le lien entre votre page et les différents réseaux sociaux, notamment Facebook, ce qui constitue un plus en terme de référencement. [En savoir plus sur les propriétés Open Graph](#).

Cette page utilise des propriétés Open Graph. Cependant, **les propriétés suivantes sont manquantes** :

- `<meta property="og:type" content="The type" />`
- `<meta property="og:url" content="http://url.com/" />`

#### Les autres conseils



## ! Google risque de ne pas utiliser votre description de la page

La page doit disposer d'une description comportant entre 20 et 300 caractères.

### Utilisation de la description par les moteurs de recherche

Lorsqu'elle est bien configurée, la description de la page se retrouve directement dans les résultats de recherche:

Amazon.fr : livres, DVD, jeux vidéo, MP3, high-tech, informatique ...

<https://www.amazon.fr/>

Achat et vente en ligne parmi des millions de produits en stock. Livraison gratuite à partir de 25€. Vos articles à petits prix : culture, high-tech, mode, jouets, sport ...

Mais les moteurs de recherche ne se servent pas nécessairement de la description que vous leur fournissez. Si celle-ci ne semble pas pertinente, ils se baseront sur d'autres critères pour présenter votre page, et vous perdrez la main sur le contenu affiché. Notez cependant que vous ne serez pas pénalisé directement par Google par exemple si votre texte est trop grand ou trop court. [En savoir plus.](#)

### Comment définir la description d'une page ?

La description de la page est renseignée grâce à la balise `<meta name="description" content="description de la page">`, qui doit être placée dans la balise `<head>`, au début du code.

! Cette page utilise Magento. Vous pouvez configurer la `<meta>` `description` via l'interface d'administration du site.

La balise `<meta>` `description` doit faire entre 20 et 300 caractères. Celle-ci en fait 873:

Légendaire et incontournable, voilà un pouf qui sait se plier à toutes vos envies !

Accessoire indispensable d'une déco design, le jumbo bag s'accorde à l'humeur du moment : ambiance cosy pour recevoir vos amis, relax sur la terrasse ou cocooning dans un coin de la chambre, il est le compagnon idéal...

## Vos principaux axes d'optimisation

0/100

#2434

**! Il manque une politique de sécurité sur la provenance de vos ressources**

Il est primordial de restreindre la provenance des contenus d'une page en vous prémunissant d'attaques de type XSS (Cross-Site Scripting).

**Les attaques XSS**

Une attaque de type XSS (Cross-Site scripting) a pour objectif d'injecter un contenu sur la page.

Pour accroître votre protection contre ces attaques, mettez en place une politique de sécurité qui va expliciter au navigateur web les serveurs qui sont autorisés à délivrer des ressources pour former la page. Si le navigateur effectue une requête vers un serveur non autorisé, il doit en informer l'internaute.

**La solution : configurer un en-tête HTTP "Content-Security-Policy" (CSP)**

Pour spécifier une politique de sécurité sur la provenance de vos ressources, vous devez configurer votre serveur afin que la réponse de la ressource principale contienne l'en-tête HTTP "Content-Security-Policy".

Voici un exemple d'utilisation de cet en-tête :

```
Content-Security-Policy: script-src 'self' https://apis.google.com
```

Exemple

Dans ce cas, la page se charge correctement à condition que tous les scripts proviennent de l'hôte courant ou de `https://apis.google.com`.

[Le lien suivant](#) vous aidera à en savoir plus sur cet en-tête HTTP. Vous serez alors amené à utiliser les directives exposées [sur ce lien](#).

**Soyez vigilant, si l'en-tête est mal configuré, certains de vos contenus, scripts, ou encore styles pourront être bloqués, ce qui pourrait engendrer des effets de bords non souhaités. De plus, les restrictions s'appliquent à toutes les pages du site.** Nous vous conseillons de tester les différentes pages de votre site avant de déployer l'instruction dans votre environnement de production.

**La CSP peut se configurer avec Apache. Assurez vous tout d'abord que le module `mod_headers` est bien activé. Ensuite, dans votre configuration (fichier `.htaccess` par exemple), vous pouvez indiquer votre politique de sécurité en adéquation avec votre cas.** Ci-dessous un exemple :

```
<IfModule mod_headers.c>
Header set Content-Security-Policy "script-src 'self' https://www.google.com"
</IfModule>
```

Exemple

Dans cet exemple, la page n'accepte que les scripts provenant du même nom de domaine et de google.com.

Aucune CSP n'a été détectée sur cette page : elle est plus facilement exposée à des attaques de type XSS.

0/100

#2521

**! 11 cookies ne sont pas sécurisés**

Un cookie transmis au navigateur via le protocole HTTP doit pouvoir transiter uniquement sur une connexion sécurisée (sauf cas particulier).

**Les cookies**

Les cookies sont placés par le serveur web sur le navigateur web via l'en-tête HTTP `Set-Cookie`. Le navigateur retransmet ensuite ces informations lors des prochaines requêtes au serveur via l'en-tête HTTP `Cookie`. Si votre serveur fournit un cookie sur une connexion sécurisée (HTTPS), c'est probablement que ce cookie contient des informations sensibles : vous devez alors garantir qu'il ne pourra pas être exploité sur une connexion non sécurisée.

**L'instruction Secure**

En ajoutant l'instruction `Secure` au niveau de l'en-tête HTTP `Set-Cookie`, le serveur informe au navigateur qu'il n'est autorisé à retransmettre le cookie que sur des requêtes sécurisées. [Lire l'article dédié à la sécurisation de cookies](#) pour en savoir plus.

**Attention : vérifiez que la redirection HTTP vers HTTPS est bien en place sur votre site. Dans le cas contraire, le cookie Secure pourrait ne pas être envoyé lors d'une requête HTTP.**

**L'en-tête HTTP `Set-Cookie` peut se configurer avec Apache. Assurez vous tout d'abord que le module**

L'en-tête HTTP **Set-Cookie** peut se configurer avec Apache. Assurez vous tout d'abord que le module **mod\_headers** est bien activé. Ensuite, vous pouvez indiquer l'en-tête dans votre configuration (fichier .htaccess par exemple), comme sur l'exemple ci-dessous :

```
<IfModule mod_headers.c>
seulement pour Apache > 2.2.4:
Header edit Set-Cookie ^(.*)$ $1;HttpOnly;Secure

versions antérieures:
Header set Set-Cookie HttpOnly;Secure
</IfModule>
```

Example

Les cookies suivants ne sont pas sécurisés, vous devriez ajouter l'instruction **Secure** lors du **Set-Cookie** :

<https://dev.jumbobag.fr/>

- Set-Cookie: frontend=e762ab627e88a63d1c50111b631d76b2; expires=Wed, 22-Nov-2017 15:18:55 GMT; Max-Age=3600; path=/; domain=dev.jumbobag.fr; HttpOnly

[payments-de.amazon.com/gp/widgets/abtests?mercha\[...\]lback=jsonp1511360339180](payments-de.amazon.com/gp/widgets/abtests?mercha[...]lback=jsonp1511360339180)

- Set-Cookie: ubid-main=260-1696586-6434069; Domain=.amazon.com; Expires=Tue, 17-Nov-2037 14:18:59 GMT; Path=

[payments-de.amazon.com/gp/apa/csm/jsonp?sellerId\[...\]27entries%27%3A%5B%5D%7D](payments-de.amazon.com/gp/apa/csm/jsonp?sellerId[...]27entries%27%3A%5B%5D%7D)

- Set-Cookie: ubid-main=260-1696586-6434069; Domain=.amazon.com; Expires=Tue, 17-Nov-2037 14:18:59 GMT; Path=

[payments-de.amazon.com/gp/apa/csm/jsonp?sellerId\[...\]27entries%27%3A%5B%5D%7D](payments-de.amazon.com/gp/apa/csm/jsonp?sellerId[...]27entries%27%3A%5B%5D%7D)

- Set-Cookie: ubid-main=260-1696586-6434069; Domain=.amazon.com; Expires=Tue, 17-Nov-2037 14:18:59 GMT; Path=

[payments-de.amazon.com/gp/apa/csm/jsonp?sellerId\[...\]27entries%27%3A%5B%5D%7D](payments-de.amazon.com/gp/apa/csm/jsonp?sellerId[...]27entries%27%3A%5B%5D%7D)

- Set-Cookie: ubid-main=260-1696586-6434069; Domain=.amazon.com; Expires=Tue, 17-Nov-2037 14:18:59 GMT; Path=

<https://payments-de.amazon.com/gp/widgets/sessionstabilizer>

- Set-Cookie: ubid-main=260-1696586-6434069; Domain=.amazon.com; Expires=Tue, 17-Nov-2037 14:18:59 GMT; Path=
- Set-Cookie: session-token=QcQyHCCPCOMi/eaHmyZg1Qd7AttiquUenOZH9XvgI9BI4eLQpFU/E0BWys8A93fBpC8RIZFYvtA07uXw6sTskP0hr4jK5rCelGPT73PFeijtU6es/pRy4hrp1Puf66CwZ3YbWoXFmIcOvMLNiZaO9MqjH2ndJJ2UH4YLrcocvBi9F53N5Y1T6n5CgANv7/xZp; Domain=.amazon.com; Expires=Tue, 17-Nov-2037 14:18:59 GMT; Path=
- Set-Cookie: session-id=262-8275357-3496934; Domain=.amazon.com; Expires=Tue, 17-Nov-2037 14:18:59 GMT; Path=
- Set-Cookie: session-id-time=2142080339; Domain=.amazon.com; Expires=Tue, 17-Nov-2037 14:18:59 GMT; Path=

[api.instagram.com/v1/users/623597756/media/recen\[...\]hee3f13f56c4b0703b.parse](api.instagram.com/v1/users/623597756/media/recen[...]hee3f13f56c4b0703b.parse)

- set-cookie: rur=ASH; Path=
- set-cookie: urlgen="{\"time\": 1511360337\\054 \\159.8.93.142\\\": 36351}:1eHVrd:bPqZVr8T2ujyAosbxypu4mxsJBM\"; Path=

**! 1 iframe peut être sécurisée avec l'attribut sandbox**

Pensez à restreindre tant que possible les actions pouvant être menées par les contenus externes intégrés à votre site.

**Contenus externes**

Les contenus externes (widgets pour les réseaux sociaux, publicités, etc), intégrés par l'intermédiaire de la balise <iframe>, doivent faire l'objet d'une attention particulière en terme de sécurité. Pour limiter les risques, le W3C a proposé l'attribut `sandbox` dans sa spécification du HTML5, qui permet de restreindre les actions possibles depuis l'iframe (sur les navigateurs récents).

**Que faire ?**

Vous devez ajouter l'attribut `sandbox` sur toutes vos iframes afin de maîtriser tant que possible le comportement des contenus externes injectés. Veillez cependant à **utiliser les bonnes valeurs pour cet attribut, comme indiqué sur cet article dédié à ce sujet.**

Il manque l'attribut sandbox sur l'iframe suivante :

- ```
<iframe src="//www.googletagmanager.com/ns.html?id=GTM-T55T8X"
height="0" width="0" style="display:none;visibility:hidden">
```

Les autres conseils

✓ Désactivez la détection automatique du type des ressources

Protégez-vous de l'exploitation malveillante du MIME Sniffing.


Le MIME-Type sniffing

Les navigateurs Internet Explorer et Chrome disposent d'une fonctionnalité dite de "MIME-Type sniffing", qui consiste à détecter automatiquement le type d'une ressource web. Ainsi, une ressource déclarée comme étant une image pourra être interprétée comme un script si tel est son contenu.

Une personne malveillante pourrait profiter d'un envoi de fichier sur votre site par exemple pour injecter du code malveillant. Nous vous conseillons de désactiver le MIME-Type sniffing pour limiter les effets de l'envoi d'un tel fichier.

La solution : configurer un en-tête HTTP "X-Content-Type-Options"

Ajoutez **dans les réponses de chacune de vos ressources** l'en-tête HTTP "X-Content-Type-Options", associé à la valeur "nosniff". Cela vous permettra de vous prémunir du risque d'une mauvaise interprétation de vos ressources.

 **L'en-tête HTTP "X-Content-Type-Options" peut se configurer avec Apache. Assurez vous tout d'abord que le module `mod_headers` est bien activé. Ensuite, vous pouvez indiquer l'en-tête dans votre configuration (fichier `.htaccess` par exemple), comme sur l'exemple ci-dessous :**

```
<IfModule mod_headers.c>
Header always set X-Content-Type-Options "nosniff"
</IfModule>
```

Exemple

Les ressources suivantes sont hébergées par un parti tiers, il se peut donc qu'elles ne soient pas sous votre responsabilité. Vous devriez cependant considérer toute alternative possible à ces fichiers pour rester en accord avec la bonne pratique.

- cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css
- <https://fonts.googleapis.com/css?family=Lato:100,300,400,700,900>
- <https://fonts.googleapis.com/css?family=Open+Sans:300,400,700>
- <https://static-eu.payments-amazon.com/OffAmazonPayments/de/lpa/js/Widgets.js>
- <https://www.googletagmanager.com/gtm.js?id=GTM-T55T8X>
- <https://widgets.trustedshops.com/js/X3B73FD19E154D0F81DD7DD1C76F880D7.js>
- api.instagram.com/v1/users/623597756/media/recent?access_token=hee3f13f56c4b0703b.parse
- scontent.cdninstagram.com/t51.2885-15/s640x640/s1500/978134161590321152_n.jpg
- scontent.cdninstagram.com/t51.2885-15/s640x640/s1500/333347435967938560_n.jpg
- scontent.cdninstagram.com/t51.2885-15/s640x640/e35/674250827016437760_n.jpg
- scontent.cdninstagram.com/t51.2885-15/sh0.08/e35/993686711937269760_n.jpg
- <https://payments-de.amazon.com/gp/widgets/sessionstabilizer>
- <https://eu.account.amazon.com/lwa/js/sdk/login1.js>
- payments-de.amazon.com/gp/apa/csm/jsonp?sellerId=...&entries%27%3A%5B%5D%7D
- payments-de.amazon.com/gp/widgets/abtests?mercha=...&lback=jsonp1511360339180
- et 15 autres

Vos principaux axes d'optimisation

0/100

#2421

! Certaines ressources sont très lourdes

Les fichiers trop lourds sont à proscrire sur une page web. Assurez vous que tous les éléments du fichier sont nécessaires au chargement de la page. Vous pouvez mettre en oeuvre plusieurs types d'actions :

- ce fichier correspond à une vidéo. Il est préférable de laisser à l'utilisateur le choix de charger cette quantité de données ou non (pas de lecture automatique par défaut)

Cette page contient **un fichier dont la taille est supérieure à 1Mo** :

- <https://dev.jumbobag.fr/media/homevideo/720p.webm> (4,74Mo)

0/100

#2443

! Cette page charge 7,81 Mo de données, diminuez ce poids

Le poids de votre page est important, ce qui ralentit sa vitesse d'affichage, d'autant plus sur les connexions bas débit. Cela peut également générer de la frustration pour vos internautes qui sont limités par des forfaits data (voir whatdoesmysitecost.com).

Situer le poids de ma page web

En février 2016, [le poids moyen des 100 pages les plus visitées dans le monde était de 1,38Mo](#).

Comment réduire le poids de ma page ?

Vous pouvez vous reporter à notre catégorie "Volume de données" pour voir quelles sont les optimisations possibles dans votre cas. De manière générale, les images sont souvent à pointer du doigt.

Veillez aussi à construire vos pages web de manière à charger d'abord ce qui est essentiel à l'internaute en priorité (optimisation du chemin critique de rendu).

Pour tout autre contenu (plugins de réseaux sociaux, publicités, contenus situés en bas de page...), il est préférable de différer le chargement (asynchrone, lazy-loading...), pour ne pas pénaliser les contenus prioritaires.

Afin de vous aider à établir vos priorités d'optimisation, voici la répartition du poids de la page par type de ressources :

- **Autres** : 63,64% du poids total
- **Images** : 22,39% du poids total
- **JavaScript** : 10,06% du poids total
- **Polices de caractères** : 1,72% du poids total
- **CSS** : 1,69% du poids total
- **Textes** : 0,46% du poids total
- **JSON** : 0,03% du poids total

Par ailleurs, voici les 10 ressources les plus lourdes qui sont téléchargées lors du chargement de votre page :

- <https://dev.jumbobag.fr/media/homevideo/720p.webm> (4973 ko)
- [static-eu.payments-amazon.com/OffAma\[...\]pa/js/Widgets.js](#) (247 ko)
- [dev.jumbobag.fr/media/js/51360117819\[...\]1bf87cb2d154e.js](#) (246 ko)
- [dev.jumbobag.fr/skin/frontend/jumbob\[...\]parallax-bg.jpg](#) (212 ko)
- <https://dev.jumbobag.fr/media/revslider/templates/slide2.jpg> (176 ko)
- [https://dev.jumbobag.fr/media/wysiwyg/fabrication-1.png](#) (143 ko)
- [scontent.cdninstagram.com/t51.2885-1\[...\]5967938560_n.jpg](#) (127 ko)
- [scontent.cdninstagram.com/t51.2885-1\[...\]1590321152_n.jpg](#) (104 ko)
- [dev.jumbobag.fr/skin/frontend/jumbob\[...\]xtrem-image.png](#) (93 ko)
- [scontent.cdninstagram.com/t51.2885-1\[...\]1937269760_n.jpg](#) (77 ko)

0/100

#2436

! 4 images sont redimensionnées côté navigateur

Une page récupérant une image plus grande que celle réellement affichée charge inutilement un volume de données trop important.

Redimensionnement d'images

Le redimensionnement d'une image côté navigateur pour en réduire la taille d'affichage n'est pas recommandé.

Fournissez directement vos images aux dimensions d'affichage utilisées sur votre site. Vous évitez ainsi l'envoi de données inutiles sur le réseau, ce qui diminue le temps de chargement de la page.

Responsive design ? Images Retina ?

Le responsive design ou encore l'utilisation d'images retina ne justifient pas un redimensionnement d'images. Même dans ces cas, différentes méthodes existent afin de délivrer vos images aux bonnes tailles. Nous vous conseillons de lire les ressources suivantes :

- [Introduction sur les images responsive](#)
- [Picturefill, pour supporter l'élément <picture> dès maintenant \(EN\)](#)
- [RICG, groupe de travail sur les images responsive \(EN\)](#)

Ne redimensionnez pas les images suivantes :

- [dev.jumbobag.fr/media/wysiwyg\[...\]tion-1.png](#) (taille d'affichage : 150x272)
- [dev.jumbobag.fr/skin/fronte\[...\]image.png](#) (taille d'affichage : 236x233)
- [https://dev.jumbobag.fr/media/wysiwyg/faq.jpg](#) (taille d'affichage : 150x460)
- [dev.jumbobag.fr/skin/fronte\[...\]image.png](#) (taille d'affichage : 252x222)

Sur cette page, 14 images sont redimensionnées de moins d'1/3 de leur taille. Si le site repose sur du Responsive Web Design, ce redimensionnement peut s'avérer cohérent. C'est pourquoi les ressources suivantes ne sont pas pénalisées au sein de ce conseil, mais nous vous invitons à y porter une vigilance particulière afin de vous assurer que leur redimensionnement n'impacte pas les performances de la page :

- [dev.jumbobag.fr/skin/fronte\[...\]image.png](#) (taille d'affichage : 160x120)
- [dev.jumbobag.fr/skin/fronte\[...\]image.png](#) (taille d'affichage : 315x289)
- [dev.jumbobag.fr/skin/fronte\[...\]image.png](#) (taille d'affichage : 327x211)
- [dev.jumbobag.fr/skin/fronte\[...\]image.png](#) (taille d'affichage : 380x203)
- [dev.jumbobag.fr/skin/fronte\[...\]image.png](#) (taille d'affichage : 104x129)
- [dev.jumbobag.fr/skin/fronte\[...\]image.png](#) (taille d'affichage : 227x187)
- [dev.jumbobag.fr/skin/fronte\[...\]image.png](#) (taille d'affichage : 246x313)
- [dev.jumbobag.fr/skin/fronte\[...\]image.png](#) (taille d'affichage : 370x46)
- [dev.jumbobag.fr/skin/fronte\[...\]image.png](#) (taille d'affichage : 264x239)
- [dev.jumbobag.fr/skin/fronte\[...\]image.png](#) (taille d'affichage : 424x203)
- [dev.jumbobag.fr/skin/fronte\[...\]image.png](#) (taille d'affichage : 147x135)
- [dev.jumbobag.fr/skin/fronte\[...\]image.png](#) (taille d'affichage : 370x370)
- [dev.jumbobag.fr/skin/fronte\[...\]image.png](#) (taille d'affichage : 169x152)
- [dev.jumbobag.fr/skin/fronte\[...\]image.png](#) (taille d'affichage : 402x306)

Les autres conseils

0/100

#2461

! Le format PNG n'est pas le plus adapté pour 1 de vos images

Le choix d'un format adapté pour une image permet de réduire son poids.

Le format PNG

Le format d'images PNG est destiné aux images nécessitant la fonctionnalité de transparence, ou encore aux petites images, disposant de peu de détails et de couleurs.

Préférez le format JPEG...

Le principal défaut du format PNG est de ne pas supporter la perte de qualité. En effet, un format tel que JPEG propose de "dégrader" la qualité de l'image sans que cela ne soit perceptible par l'utilisateur. Vous pouvez ainsi diminuer la qualité de l'image d'[environ 25%](#) sans que vos utilisateur ne s'en rendent compte.

Une image avec un poids important sera donc sans doute mieux compressée en utilisant le format JPEG.

...ou PNG-8

Dans le cas où votre image nécessite obligatoirement d'utiliser le mécanisme de transparence, non supporté par le format JPEG, vous devriez convertir votre image PNG "classique", en PNG-8. Ce format, basé sur une palette de 256 couleurs maximum permet de diminuer le poids de l'image sans altérer significativement sa qualité en général. Des outils tels [pngquant](#) ou encore [tinypng](#) vous aideront dans cette démarche.

Enfin, si la qualité proposée par le format PNG-8 n'est pas satisfaisante pour votre image, vous pouvez vous renseigner sur [les techniques avancées possibles \(EN\)](#) pour obtenir ce comportement sans délivrer une image au format PNG. Par exemple, il est possible de séparer votre image en 2 images JPEG, l'une contenant les données de transparences et l'autre contenant les données liées aux couleurs, et de regrouper l'image côté client à l'aide d'un élément CANVAS.

Pour en savoir plus, n'hésitez pas à consulter [cet article sur la compression des images \(EN\)](#).

Il semble que le format JPEG soit plus adapté pour les images suivantes :

- <https://dev.jumbobag.fr/media/wysiwyg/fabrication-1.png> (142ko)

70/100

#2500

! Vos fontes de caractères peuvent être allégées

Veillez à ce que les formats de fontes utilisés par le navigateur soient les plus légers.

Indiquer plusieurs types de fontes

Les différents navigateurs web (Firefox, Chrome, Safari, Internet Explorer, Opera...) ne supportent pas tous les mêmes formats de fontes. Votre objectif est d'utiliser le format le plus léger possible pour chaque plateforme.

Lorsque vous utilisez une fonte, spécifiez ainsi plusieurs formats, via l'instruction CSS `@font-face` : le navigateur utilisera le premier format compatible sur lequel il tombe.

Que faire ?

Repassez sur les instructions `@font-face` qui font défaut, et vérifiez si l'ordre indiqué est cohérent, ou encore si aucune erreur de syntaxe n'est présente. Voici un article sur le sujet afin de vous aider à [déterminer le bon ordre à choisir pour vos fontes de caractères](#). Plus succinctement, **l'ordre à respecter pour vos formats est le suivant : EOT, WOFF2, WOFF, TTF, puis SVG.**

Les ressources suivantes sont récupérées, cependant vous devriez envisager l'utilisation du format WOFF2 à la place :

- [dev.jumbobag.fr/skin/frontend/base/default/nwidth\[...\]/ns/revicons.woff?5510888](https://dev.jumbobag.fr/skin/frontend/base/default/nwidth[...]/ns/revicons.woff?5510888)
- <https://widgets.trustedshops.com/fonts/tsfonts.woff>

95/100

#2364

✓ Vous pouvez réduire certains fichiers HTML (minification)

En compressant votre code HTML (y compris le code JavaScript et CSS intégré), vous pouvez libérer de nombreux octets de données et réduire les délais de téléchargement et d'analyse.

Réduisez la taille des ressources HTML suivantes afin de gagner 5.5 Ko (réduction de 23%).

- Une réduction de la taille de <https://dev.jumbobag.fr/> pourrait libérer 5.5 Ko (réduction de 23%).

95/100

#2387

✓ Vous pouvez réduire certains fichiers CSS (minification)

En compressant votre code CSS, vous pouvez libérer de nombreux octets de données et réduire les délais de téléchargement et d'analyse.

Réduisez la taille des ressources CSS suivantes afin de gagner 7.9 Ko (réduction de 12%).

- Une réduction de la taille de [dev.jumbobag.fr/me\[...\].css](dev.jumbobag.fr/me[...].css) pourrait libérer 7.9 Ko (réduction de 12%).

De nombreux outils existent pour minifier des fichiers CSS. C'est le cas de [YUI Compressor](#) ou [cssmin.js](#), recommandés par Google.

Si vous souhaitez en savoir plus sur [la minification CSS](#), n'hésitez pas à consulter cet article sur le sujet !

99/100

#2384

✓ Activez la compression des ressources

En compressant vos ressources avec "gzip" ou "deflate", vous pouvez réduire le nombre d'octets envoyés sur le réseau.

Autorisez la compression des ressources suivantes afin de réduire le volume de données transférées de 169.6 Ko (réduction de 70%).

Les ressources suivantes sont hébergées par un parti tiers, il se peut donc qu'elles ne soient pas sous votre responsabilité. Vous devriez cependant considérer toute alternative possible à ces fichiers pour rester en accord avec la bonne pratique.

- La compression de [static-eu.payments\[...\].js](static-eu.payments[...].js) pourrait libérer 169.6 Ko (réduction de 70%).

🚩 Cette page est retournée par un serveur Apache. Vérifiez s'il utilise bien le module [mod_deflate](#).

Vos principaux axes d'optimisation

0/100

#74

! Évitez d'exposer votre version d'Apache

Les réponses HTTP de votre serveur contiennent un en-tête HTTP `Server` qui indique la version de votre serveur Apache: `Apache/2.2`. Il sera plus difficile pour une personne malveillante d'attaquer votre site si elle ne connaît pas la version de votre serveur Apache. Pensez à modifier cette valeur.

Par exemple, si vous êtes sur un serveur Linux, éditez votre fichier `/etc/apache2/conf.d/security`. Vous devez alors modifier les valeurs de `ServerSignature` et de `ServerTokens`, comme indiqué ci-dessous:

```
# Cacher la version dans l'en-tête HTTP 'Server'.
# exemple: afficher uniquement "Server: Apache"
ServerTokens Prod
```

Exemple

```
# Ne pas ajouter de ligne contenant
# le nom et la version du serveur
# à la fin de chaque réponse.
ServerSignature Off
```

Exemple

Cependant, gardez en tête que le meilleur moyen de se prémunir d'attaques est de régulièrement mettre à jour votre serveur Apache.

Vos principaux axes d'optimisation

0/100

#2302

! Évitez les sélecteurs jQuery trop spécifiques

Vous utilisez des sélecteurs jQuery trop spécifiques qui diminuent les performances de votre site web : [en savoir plus](#). Voici un exemple de bonne utilisation de jQuery :

```
$( ".data table.firstClass td.secondClass" );  
// Plus performant: supprimer le sélecteur  
// intermédiaire si possible  
$( ".data td.secondClass" );
```

Exemple

Si possible, **simplifiez les sélecteurs suivants** :

[dev.jumbobag.fr/skin/frontend/jumbobag/\[...\]ault/js/produit.js](#)

- \$('tissu p .fa')
- \$('tissu p .fa')
- \$(".tissu p .fa")
- \$('tissu p .fa')
- et 9 autres

0/100

#2301

! Préférez le sélecteur jQuery \$("#id").find("test") plutôt que \$("#id test")

Vos sélecteurs débutent par des identifiants (#id), c'est une bonne chose: jQuery sélectionne ainsi les éléments via `document.getElementById()`, qui est le sélecteur le plus performant.

Cependant, les autres sélections informées après l'ID devraient être placées dans un `find()`.

```
// fast  
$( "#container div.element" )  
// super-fast  
$( "#container" ).find( "div.element" );
```

Exemple

Les fichiers suivants **contiennent des sélecteurs commençant par des identifiants, suivis par d'autres éléments** :

Ressources provenant de jumbobag

[dev.jumbobag.fr/skin/frontend/jumbobag/\[...\]ault/js/produit.js](#)

- \$("#fond-pres-produit img")
- \$("#fond-entretien img")
- \$("#product-options-wrapper label.required")
- \$("#shopper_gallery_carousel li")

Ressources provenant de partis tiers

[images-na.ssl-images-amazon.com/images/\[...\]ib_V494532994_js](#)

- jQuery("#expirationmonthandyear select")
- jQuery("#expiryUpdateForm select")

! Évitez de manipuler le DOM dans des boucles avec jQuery

Travailler avec le DOM a un coût. Si vous avez besoin d'ajouter des éléments à un nœud, il est préférable de les ajouter tous d'un coup plutôt qu'un par un.

Vous avez **1 manipulation(s) du DOM dans des boucles** :

[dev.jumbobag.fr/media/js/5136011781963d\[...\]b61bf87cb2d154e.js](https://dev.jumbobag.fr/media/js/5136011781963d[...]b61bf87cb2d154e.js)

- ```
for(var e=0;e<b.slideamount;e++){var f=Math.round(Math.random()*b.slideamount);b.ul.find(">li:eq("+f+")").prependTo(b.ul)}
```