

Audit d'accessibilité simplifié - www.syvicol.lu

Date : 18 novembre 2022

Appréciation générale

Niveau d'accessibilité global pour les critères testés : **faible**.

(Échelle : très faible, faible, moyen, bon, très bon)

Avertissement

Attention, l'audit effectué est un audit simplifié et non un audit de conformité (ou audit "complet"). Il a vocation à détecter une série de problèmes d'accessibilité mais n'est pas exhaustif. Le fait qu'aucun problème ne soit remonté pour un critère d'accessibilité donné ne signifie pas qu'il n'y a pas de problème d'accessibilité pour ce critère. De même, lorsque nous rapportons une occurrence d'un problème, ce problème peut avoir d'autres occurrences. Il est nécessaire de vérifier de manière exhaustive l'accessibilité de ce site conformément au référentiel [RGAA](#).

Échantillon de pages et référentiel

Voici les pages qui ont été évaluées lors de cet audit :

- Page P01 : [Accueil - Syvicol](#)
- Page P02 : [Contact - Syvicol](#)
- Page P03 : [BETTENDORF - Syvicol](#)

Méthode d'évaluation : [Méthode de contrôle simplifiée de l'accessibilité pour le Luxembourg – v1.2](#)

Référentiel : [RGAA v4.1](#)

Déclaration sur l'accessibilité

Le site ne comporte pas de **déclaration sur l'accessibilité**. Celle-ci est obligatoire d'après l'article 5 de la [loi du 28 mai 2019](#). Cette déclaration s'effectue après avoir réalisé un audit de conformité basé sur le [RGAA](#). Pour créer une déclaration sur base des résultats d'un audit de conformité, le [formulaire disponible à cet effet sur accessibilite.lu](#) peut être utilisé. Une fois la déclaration d'accessibilité publiée, l'éditeur du site a 30 jours pour en informer le SIP par mail à l'adresse accessibilite@sip.etat.lu.

Documents bureautiques en téléchargement

L'accessibilité d'un échantillon de 980 fichiers bureautiques en téléchargement sur le site considéré a été analysée. Le niveau d'accessibilité global présenté dans la section "Appréciation générale", doit être nuancé au vu

des points suivants :

Un nombre important de contenus dérogés impacte la pertinence des résultats de l'audit (la liste des dérogations prévues par la loi est disponible sur [la page "obligations" du site accessibilite.lu](#)). Le site contient un grand nombre de contenus dérogés. En effet, un nombre important de fichiers PDF a été publié avant le 23 septembre 2018 (soit 311 fichiers). Bien que ces fichiers soient exemptés d'obligation de mise en accessibilité par la loi du 28 mai 2019, il est recommandé de les rendre accessibles, sans quoi une part importante du contenu mis à disposition sur ce site ne pourra être consulté par les personnes en situation de handicap.

Une quantité importante de documents bureautiques en téléchargement a été détectée, ce qui peut avoir un impact négatif sur l'accessibilité globale de ce site. Sur l'échantillon de fichiers analysés, 99 % sont des documents PDF et 1 % sont des formulaires PDF. Le format PDF est en général moins accessible que les pages Web et que les documents Office (.docx, .pptx, etc.); l'accessibilité des formulaires est particulièrement importante dans la mesure où ils sont en général nécessaires à la réalisation de procédures administratives.

Sur les fichiers PDF qui entrent dans le cadre de la loi (publication après le 23 septembre 2018) et issus de l'échantillon, 99 % ont au moins un problème d'accessibilité (ex: absence de titre, absence de langue par défaut, absence de bookmarks). Ces informations sont données à titre indicatif, car la présence d'une alternative accessible n'a pas été vérifiée dans le cadre de cet audit simplifié.

Pour information, le SIP met à disposition une liste de [critères d'accessibilité pour les documents bureautiques en téléchargement](#). Pour chaque document bureautique en téléchargement, il est possible de le rendre accessible en respectant les critères mentionnés précédemment, ou de proposer une alternative accessible, sous la forme d'une page Web ou d'un document Office (.docx, .pptx, etc.) proposant les mêmes informations.

Annexe technique

Thématique "images"

Recommandations générales

Donner à chaque [image porteuse d'information](#) une [alternative textuelle](#) pertinente et une [description détaillée](#) si nécessaire. Lier les [légendes](#) à leurs images. Remplacer les [images textes](#) par du [texte stylé](#) lorsque c'est possible. Pour trouver la bonne alternative textuelle pour une image donnée, vous pouvez vous aider de l'[arbre de décision proposé par la WAI](#).

Cas rencontré : images de décoration

Les [images de décoration](#) ne doivent pas avoir d'alternative textuelle et doivent être ignorées correctement par les technologies d'assistance. Dans le cas contraire, elles peuvent causer des problèmes de compréhension aux utilisateurs de lecteurs d'écran (aveugles et grands malvoyants). Pour une image *bitmap*, son attribut `alt` doit être vide. Pour une image vectorielle SVG, celle-ci doit avoir l'attribut `aria-hidden="true"`.

Exemple de problème détecté sur la [page P01](#)

Le logo Syvicol en pied de page devrait recevoir un attribut `alt=""`.

Thématique "couleurs"

Recommandations générales

Ne pas donner l'[information](#) uniquement par la couleur et utiliser des [contrastes](#) de couleurs suffisamment élevés pour les textes, les [composants d'interface](#) ou les éléments porteurs d'informations.

Cas rencontré : informations données uniquement par la couleur

Ce type d'information est un problème pour les personnes déficientes visuelles, aveugles ou par exemple les personnes qui ne voient pas certaines couleurs ou ne perçoivent simplement pas les couleurs. Pour chaque information véhiculée par la couleur, il est nécessaire de mettre en place une alternative, comme par exemple un changement de style (graisse du texte, taille du texte, soulignement, etc.)

Exemple de problème détecté sur la [page P01](#)

Le changement de couleur constitue le seul signe distinctif de l'article de menu actif. Un soulignement, un changement de graisse, etc. permettrait aux personnes souffrant d'anomalies de la vision des couleurs de se repérer. Absence d'un attribut `aria-current`.

Cas rencontré : contrastes des textes

Les contrastes de couleurs sont importants pour plusieurs types de déficiences visuelles comme celles des grands malvoyants ou des personnes ayant des problèmes de perception des couleurs. Les contrastes minimaux d'un texte sur le fond de page sont définis par le RGAA comme suit :

- Pour un texte sans effet de graisse
 - De taille inférieure à 24 px : le contraste minimum est de 4.5:1 ;
 - De taille supérieure ou égale à 24 px : le contraste minimum est de 3:1.
- Pour un texte en gras
 - De taille inférieure à 18.5 px : le contraste minimum est de 4.5:1 ;
 - De taille supérieure à 18.5 px : le contraste minimum est de 3:1.

Pour vérifier les contrastes, on peut utiliser un outil tel que [« Colour Contrast Analyser »](#). Si les problèmes de contraste ne peuvent être résolus simplement, il est aussi possible d'utiliser un *« style switcher »*. C'est une fonctionnalité proposée par le site qui permet de renforcer les contrastes pour les personnes qui ont des problèmes avec les couleurs. Un exemple de *« style switcher »* est disponible sur le site [snfc.com](#), dans son menu *« Accessibilité »*.

Exemple de problème détecté sur la [page P01](#)

Certains textes sur cette page n'ont pas un contraste suffisant (p. ex. les articles de menu sur fond bleu : 3.77:1 contre un contraste de couleur attendu de 4.5:1 ; les hyperliens rouges 4.45:1 contre 4.5:1 min. requis ; les étiquettes "Développement durable" ou "Enseignement fondamental" 1.73:1 contre 4.5:1 min. requis).

Thématique "tableaux"

Recommandations générales

Associer correctement les [tableaux de données à leur titre](#), donner à chaque [tableau de données complexe](#), un [résumé](#), identifier clairement les cellules d'en-tête, utiliser un mécanisme pertinent pour lier les cellules de données aux [cellules d'en-tête](#). Pour chaque [tableau de mise en forme](#), veiller à sa bonne linéarisation.

Cas rencontré : déclaration des entêtes et liaison des cellules d'entêtes et de données

Les utilisateurs de lecteurs d'écran ou de loupe d'écran vocalisée ne peuvent percevoir un tableau dans son ensemble. Il est donc important de leur communiquer des informations de contexte sur chaque cellule, notamment à quelles entêtes chaque cellule est reliée. Ces informations peuvent être données via des structures HTML dédiées. Les entêtes de colonnes et de lignes doivent notamment être déclarées via la balise `<th>`.

Dans un tableau de données simple, où chaque entête est valable pour l'ensemble de la ligne ou de la colonne, la relation entre les cellules et les entêtes doit être définie en appliquant un attribut `scope="col"` à toutes les entêtes de colonnes et `scope="row"` à toutes les entêtes de lignes.

Dans un [tableau de données complexe](#), chaque entête doit avoir un identifiant déclaré via l'attribut `id` et chaque cellule doit faire référence à ces entêtes via l'attribut `headers` (liste d'identifiants séparés par des espaces).

Exemple de problème détecté sur la [page P03](#)

Dans les tableaux présents sur cette page, les entêtes de lignes ne sont pas structurés via des balises `<th>`.

Thématique "liens"

Recommandations générales

Utiliser des [intitulés](#) de [liens](#) explicites, grâce à des informations de [contexte](#) notamment.

Cas rencontré : présence d'intitulés

Un lien doit toujours avoir un intitulé. Celui-ci permet aux utilisateurs de comprendre la destination du lien. La présence d'intitulés sur tous les liens est particulièrement importante pour les aveugles et malvoyants. Si un lien n'a pas d'intitulé, les lecteurs d'écran vont le lire simplement « lien » sans annoncer sa destination.

- Pour un lien qui ne contient qu'une image, l'intitulé du lien est l'alternative de l'image. Pour une image *bitmap*, son attribut `alt` doit contenir l'intitulé du lien (ex: ``). Pour une image vectorielle SVG, celle-ci doit avoir l'attribut `role="img"` et l'intitulé du lien pourra être fourni via les attributs `aria-label` ou `aria-labelledby` (ex: `<svg role="img" aria-label="facebook">...</svg>`).
- Lorsqu'un lien contient du texte et des images, l'intitulé du lien correspond à la concaténation des intitulés de ces différents éléments contenus dans le lien. (ex: pour ` Facebook` l'intitulé du lien est « logo facebook facebook ». Dans cet exemple, cet intitulé n'est pas pertinent et l'alternative de l'image doit être vide.)

Exemple de problème détecté sur la [page P01](#)

De nombreux liens image n'ont pas d'intitulé. Exemples : liens liés aux réseaux sociaux, liens de téléchargement.

Thématique "scripts"

Recommandations générales

Donner si nécessaire à chaque [script](#) une [alternative](#) pertinente. Avertir ou permettre le contrôle des scripts qui initient un [changement de contexte](#). Rendre possible le contrôle de chaque code [script](#) au moins [par le clavier et par tout dispositif de pointage](#) et s'assurer de leur [compatibilité avec les technologies d'assistance](#) notamment pour les [messages de statut](#).

Cas rencontré : éléments interactifs inaccessibles au clavier

Les composants riches développés en JavaScript doivent respecter des modèles de conception spécifiques pour pouvoir être considérés comme accessibles (modèles de conception décrits dans le document [« WAI-ARIA Authoring practices »](#)). Sans cela le composant ne sera pas correctement restitué aux utilisateurs de lecteurs d'écran qui ne sauront pas comment l'utiliser. Ces composants doivent notamment utiliser des interactions au clavier spécifiques, sans lesquelles ils seront inutilisables pour les utilisateurs de la navigation au clavier.

Exemple de problème détecté sur la [page P01](#)

La touche Espace aussi doit pouvoir afficher un sous-menu. Ce dernier doit pouvoir disparaître grâce à la touche Echapement. Quand le formulaire de recherche est affiché, la tabulation ne doit pas pouvoir intercepter des éléments non visibles.

Thématique "éléments obligatoires"

Recommandations générales

Vérifier que dans chaque page Web, le code source généré respecte les [règles d'écriture](#) correspondant au [type de document](#), que le [titre](#) est pertinent et la [langue par défaut](#), indiquée. Vérifier que les balises ne sont pas utilisées [uniquement à des fins de présentation](#), que les [changements de langues](#) et de direction de [sens de lecture](#) sont indiqués.

Cas rencontré : validité du code source de la page

Les lecteurs d'écran s'appuient sur la sémantique des balises HTML pour restituer le contenu et naviguer, si la syntaxe HTML n'est pas respectée ou si les balises sont détournées, cela peut avoir un impact sur la navigation qui peut devenir inopérante et la restitution qui peut devenir incompréhensible. Pour vérifier la validité de la syntaxe HTML, on peut utiliser le [validateur HTML du W3C](#) avec le *bookmarklet* [« WCAG 2.1 parsing error »](#). Ce *bookmarklet* filtre les résultats et ne montre que les erreurs ayant un impact sur l'accessibilité.

Exemple de problème détecté sur la [page P01](#)

Vérifier que la valeur d'attribut id de chaque élément actif est unique

Documentation : <https://dequeuniversity.com/rules/axe/4.3/duplicate-id-active?application=webdriverjs&lang=fr>

Occurrences :

- Corriger l'un des éléments suivants : Le document comporte ou un plusieurs éléments actifs partageant la même valeur d'attribut id : fieldEmail (Ex : .padding-2.align-start.desc-container > .js-cm-form[method="post"] > .js-cm-email-input[name="cm-kuluiui-kuluiui"][type="email"],)

Vérifiez que le code html de la page est valide

Valdateur : <https://validator.w3.org/nv/>

Occurrences :

- Balise de fin non trouvée "link".
- L'élément "div" n'est pas autorisé comme enfant de l'élément "span"
- Élément non fermé "span"
- Balise de fin non trouvée "span".
- Duplication de l'attribut "id".
- Pas d'élément "p" dans la portée mais une balise de fin "p" vue.
- Duplication de l'attribut "type".

Cas rencontré : indication de langue

Les lecteurs d'écran utilisent les indications de langue pour vocaliser correctement le contenu. La langue principale de la page est spécifiée via l'attribut `lang` sur l'élément `<html>`. Lorsqu'un mot d'origine étrangère est inséré dans du contenu écrit dans la langue principale de la page, il doit posséder si nécessaire une indication de langue. L'indication de langue se fait par l'intermédiaire de l'attribut `lang`. Il existe néanmoins des exceptions :

- Lorsqu'il s'agit d'un nom, l'indication de langue doit être faite uniquement quand le nom doit se prononcer dans sa langue d'origine ;
- Lorsqu'il s'agit d'un mot d'origine étrangère, présent dans le dictionnaire de la langue principale de la page, l'indication de langue n'est pas nécessaire ;
- Lorsqu'il s'agit d'un mot d'origine étrangère d'usage courant, mais absent du dictionnaire, l'indication de langue doit être faite uniquement si la prononciation dans la langue principale de la page est problématique.

Exemple de problème détecté sur la [page P01](#)

Cette page contient des textes en d'autres langues non marqués par un changement de langue (p. ex. "Zesumme spueren", "Éislek am neie Liicht", "Newsletter").

Thématique "structuration de l'information"

Recommandations générales

Utiliser des [titres](#), des [listes](#), et des citations pour structurer l'information. S'assurer que la structure du document est cohérente.

Cas rencontré : titres

Les [titres](#) permettent aux utilisateurs de comprendre la structure du document. Ils permettent aussi aux utilisateurs aveugles, malvoyants et handicapés moteurs de naviguer de titre en titre dans la page. Il est donc important d'avoir une bonne hiérarchie de titres dans une page. Cette hiérarchie est similaire à un sommaire de document. La vérification de la hiérarchie de titres peut se faire via l'extension de navigateur [HeadingsMap](#).

Exemple de problème détecté sur la [page P01](#)

Les titres de pied de page "Sitemap" etc. sont présentés par des extensions telles que headingsMap ou Wave comme imbriqués sous "Contactez-nous" alors qu'il s'agit de blocs différents sur la page.

Cas rencontré : structure du document HTML5

La structuration du document HTML5 permet aux aveugles, grands malvoyants et handicapés moteurs de naviguer très rapidement entre les zones principales de la page (header, footer, zone de contenu principale, navigation, ...)

Exemple de problème détecté sur la [page P01](#)

La page n'est structurée par aucun des éléments `<main>`, `<footer>` ou `<nav>`.

Thématique "présentation de l'information"

Recommandations générales

Utiliser des [feuilles de styles](#) pour [présenter de l'information](#). S'assurer que l'information reste [compréhensible](#) lorsque les feuilles de styles sont désactivées. Vérifier l'effet de l'agrandissement à 200 % de la [taille des caractères](#) et de la redéfinition des propriétés d'espacement sur la lisibilité. S'assurer que les liens sont correctement identifiables, que la [prise de focus](#) est signalée et que l'utilisateur a le contrôle des contenus additionnels qui deviennent visibles au survol ou au focus. S'assurer que les [contenus cachés](#) sont ignorés par les technologies d'assistance et que l'information n'est pas donnée uniquement par la [forme](#), [taille ou position](#) d'un élément.

Cas rencontré : visibilité du focus

Les handicapés moteurs qui naviguent au clavier utilisent l'[indicateur de focus](#) fourni par le site sur les éléments interactifs pour savoir où ils se situent dans la page. L'indicateur de focus se déplace via les touches `tab` et `shift-tab`. L'indicateur de focus par défaut peut être désactivé via CSS, dans ce cas il est nécessaire de changer le style de l'élément interactif pour rendre l'indicateur de focus visible (sa couleur devra avoir un contraste minimum de 3:1 avec l'arrière-plan contigu).

Exemple de problème détecté sur la [page P01](#)

Dans de nombreux éléments de cette page, le focus n'est pas visible. Exemples : boutons "Annuaire des communes", "Annuaire des syndicats", items disposés sous "Événements à venir", ou encore le fait de tabuler "en aveugle" dans le menu burger caché en mode desktop.

Thématique "formulaires"

Recommandations générales

Pour chaque formulaire, associer chacun de ses [champs](#) à son [étiquette](#), grouper les [champs de même nature](#) et leur donner une [légende](#), structurer les listes de choix de manière pertinente, donner à chaque [bouton](#) un intitulé explicite. Vérifier la présence de suggestions lors des erreurs de saisie, s'assurer que le [contrôle de saisie](#) est accessible, que la finalité des champs peut être déduite et que l'utilisateur peut garder le contrôle sur ses données à caractère financier, juridique ou personnel.

Cas rencontré : identification des champs, des contrôles et des regroupements de formulaires

L'identification de ces éléments fournit aux aveugles et grands malvoyants les informations nécessaires pour pouvoir remplir un formulaire. Les handicapés moteurs utilisant un système de reconnaissance vocale s'appuient aussi sur ces éléments pour se déplacer dans les différents [champs](#) et actionner les boutons. Pour associer une [étiquette](#) (`<label>`) à un champ de formulaire on peut utiliser l'attribut `for` de l'étiquette dont la valeur doit être identique à celle de l'attribut `id` du champ. Les [champs de même nature](#) doivent être regroupés, cela peut être réalisé via l'élément `<fieldset>` auquel on doit fournir une [légende](#) via l'élément `<legend>` .

Exemple de problème détecté sur la [page P01](#)

L'étiquette du champ de recherche n'est pas visible. Un placeholder n'est ici pas suffisant, car il ne sera plus visible une fois que l'utilisateur aura commencé à remplir le champ. Une solution peut être de placer un attribut `title` sur le champ, qui contient le contenu de l'étiquette. Ce contenu sera visible au "hover", même pendant la saisie.

Cas rencontré : intitulés de boutons pertinents

Les boutons doivent avoir des intitulés pertinents, qui communiquent l'action réalisée à l'activation. Ces informations sont particulièrement importantes pour les déficients visuels et cognitifs.

Exemple de problème détecté sur la [page P01](#)

L'intitulé du bouton permettant de fermer la fenêtre de recherche n'est pas pertinent : "Close search form". Il serait nécessaire de le traduire dans la langue de la page.

Cas rencontré : contrôle et aide à la saisie

Que ce soit pour les handicapés cognitifs ou pour les aveugles et déficients visuels, il est nécessaire d'explicitier les formats de données attendus dans les consignes et les messages d'erreur, ainsi que le caractère obligatoire de la saisie de certains champs.

Exemple de problème détecté sur la [page P02](#)

Le type de données attendu dans le champ « e-mail » n'est pas précisé, et cela ni dans l'étiquette, ni dans le message d'erreur. Il est nécessaire de donner un exemple.

Thématique "navigation"

Recommandations générales

Proposer au moins deux [systèmes de navigation](#) différents dans un [ensemble de pages](#) ([menu de navigation](#), [plan du site](#) ou [moteur de recherche](#)). Donner la possibilité d'éviter ou d'atteindre les principaux regroupements de contenus en particulier la zone de contenu principale via un [lien d'évitement ou d'accès rapide](#). S'assurer que l'[ordre de tabulation](#) est cohérent et que la page ne comporte pas de piège au clavier. S'assurer que les [raccourcis clavier](#) n'utilisant qu'une seule touche sont contrôlables par l'utilisateur.

Cas rencontré : *landmarks* ARIA

Les utilisateurs aveugles utilisent pour naviguer rapidement dans une page des points de repères ou [landmarks](#). Ceux-ci définissent les principales zones de la page comme l'entête, le menu de navigation, la zone de contenu principale, le pied de page, le moteur de recherche. Chacune de ces zones doit avoir un attribut `role` dont la valeur correspond au type de zone :

- `role=banner` pour l'entête,
- `role=navigation` pour le menu de navigation,
- `role=main` pour la zone de contenu principale,
- `role=contentinfo` pour le pied de page,
- `role=search` pour le moteur de recherche.

Exemple de problème détecté sur la [page P01](#)

La page ne contient aucun rôle ARIA de type "banner", "main", "contentinfo" ou "navigation". Par ailleurs, le rôle "search" n'est pas pertinent dans la mesure où le moteur ne recherche pas dans l'intégralité du site. Ainsi, il n'a pas été possible de trouver des extraits figurant dans la page "Politique de confidentialité".

Cas rencontré : liens d'accès rapide

Les [liens d'accès rapide](#) sont des liens présents en début de page et permettent aux utilisateurs qui naviguent au clavier et aux utilisateurs malvoyants qui utilisent une loupe d'écran d'éviter des zones de contenus redondants comme l'entête et la navigation. Il est indispensable d'avoir au moins un lien d'accès rapide vers la zone de contenu principale (`<main>`). Ces liens peuvent être positionnés hors écran et apparaître à la prise de focus.

Exemple de problème détecté sur la [page P01](#)

Aucun lien d'évitement.

Cas rencontré : ordre de tabulation

L'[ordre de tabulation](#), c'est-à-dire l'ordre dans lequel le focus se déplace lorsqu'on utilise les touches `tab` et `shift-tab` doit être cohérent. Si l'ordre de tabulation est incohérent, les aveugles, les déficients visuels, les handicapés moteurs et handicapés cognitifs auront de grandes difficultés à naviguer dans la page.

Exemple de problème détecté sur la [page P01](#)

L'ordre de tabulation n'est pas cohérent. Les éléments de la boîte cookies ne doivent pas figurer parmi les derniers à recevoir le focus.