



Accessibilité

<http://www.....fr/>

Note de cadrage

Marina Korzun Sacilotto
Avril-Mai 2017

Sommaire

Introduction	3
Publics visés	4
Types de handicaps	4
Technologies d'assistance	5
Accessibilité du web	6
Les démarches reconnues en matière d'accessibilité web	6
Confort de lecture	6
Euracert	7
AccessiWeb	7
WCAG (1.0 et 2.0)	8
RGAA: une obligation pour les sites web publics	9
RGAA: référentiel technique	11
Éléments Obligatoires	11
Structuration de l'information	12
Présentation de l'information	12
Multimédia	14
Vidéos/Audio	15
Cadres	16
Scripts	17
Couleurs	17
Tableaux	18
Liens	18
Formulaires	19
Navigation	20
Consultation	21
ARIA. Construction de la sémantique	24
Recommandations pour le site de la GN (grands publics)	28
Outils d'analyse et des tests	40
Terminologies	42
Références	42
Annexe 1: Proposition Arborescence	44
Annexe 2: Proposition Page Accueil	45
Annexe 3: Proposition Gabarit des Pages	46

Introduction

Le 7 avril 1997, l'initiative pour l'accessibilité du Web, appelée aussi la WAI (Web Accessibility Initiative), était officiellement lancée par le W3C (World Wide Web Consortium). La principale mission de la WAI est de proposer des solutions techniques afin de rendre le Web accessible aux personnes handicapées et des seniors. En 1999, les recommandations pour l'accessibilité des contenus Web, WCAG (Web Content Accessibility Guidelines), sont développées.

En France, les sites web publics doivent se mettre en conformité en appliquant les règles du Référentiel Général d'Accessibilité pour les Administrations (RGAA). Le RGAA reprend la structure des recommandations internationales W3C, WAI et WCAG. Il inclut aussi les tests des contrôles afin de vérifier la conformité d'un contenu web.

Le RGAA est utilisé dans les contextes suivants :

- La mise en œuvre d'un nouveau site.
Cette approche consiste à prendre en compte les recommandations du RGAA du début à la fin du projet de création du site (analyse, production, mise en ligne, maintenance, vie du site).
- L'amélioration d'un site existant et la réalisation de la déclaration de conformité (refonte).
Cette approche consiste à évaluer l'accessibilité d'un site déjà en production, de façon à extraire des indicateurs, des résultats globaux, des recommandations opérationnelles, et des résultats pour le site et pour ses pages.

L'objectif de ce document est d'expliquer comment adapter le site internet de la gendarmerie nationale aux normes d'accessibilité (RGAA) .

Ce document s'adresse aux:

- professionnels du Web travaillant au
- référents accessibilité de la;
- administrateurs Web.

Ce document a été préparé dans le cadre d'un stage court (5 semaines). Il s'agit d'une note de cadrage. Les améliorations et les modifications sont à prévoir.

Publics visés

Types de handicaps

Les types de handicaps qui affectent l'accès des personnes au Web:

Handicaps visuels

Le handicap visuel ne concerne pas que les personnes non voyantes. Il regroupe également tout un ensemble de troubles visuels ou de maladies qui vont d'une baisse plus ou moins forte de la vue, à une vue tronquée ou altérée par des taches, par exemple, en passant par une perception des couleurs qui peut être déficiente.

Handicap auditif

La problématique d'accessibilité web aux personnes atteintes de troubles auditifs va au-delà de la restitution des sons. 80% des sourds de naissance sont illettrés. Concevoir des sites web accessibles aux sourds implique donc de repenser non seulement les vidéos mais aussi les textes.

Handicap moteur

Ce type de handicap est très répandu et peut toucher n'importe qui en quelques secondes. De nombreuses personnes peuvent, par exemple, avoir des difficultés à utiliser une souris, que ce soit en raison d'un manque de dextérité ou d'une paralysie partielle ou totale.

Handicap cognitif

Ce type de handicap concerne notamment les personnes souffrant de trouble de la concentration, de dyslexie, d'épilepsie ou encore de troubles de la mémoire. Concevoir des sites cognitivement accessibles implique de repenser l'interface de navigation, de respecter les mécanismes de perception et de compréhension traditionnels.

Technologies d'assistance

Les nouvelles technologies permettent aux personnes handicapées d'accéder à Internet. Il est nécessaire pour cela qu'elles possèdent un poste adapté en fonction de leur(s) handicap(s).

Déficiences visuelles

Des aides techniques logicielles et matérielles permettent aux personnes malvoyantes et aveugles d'accéder aux sites Internet qui sont accessibles grâce à l'utilisation :

- de logiciels de grossissement de caractères;
- des différents raccourcis clavier selon les logiciels;
- des lecteurs d'écran;
- de plages Braille;
- d'une synthèse vocale.

Déficiences auditives

Beaucoup de personnes sourdes ont des difficultés de lecture, ce qui rend difficile leur consultation des sites Internet. Elles ont besoin de communiquer en Langue des Signes (webcam, vidéos, visioconférences, etc). A contrario, beaucoup de personnes malentendantes ne signent pas et ont besoin d'un accès au texte ou d'une amplification du son pour les vidéos ou contenus audio en ligne.

Déficiences motrices

Les personnes en situation de handicap moteur ont donc une atteinte partielle ou totale de la mobilité, notamment des membres supérieurs ou/et inférieurs, et parfois une difficulté à communiquer. Du fait de leur déficience, elles peuvent éprouver des difficultés à accéder et à utiliser leur poste de travail informatique. Des aides techniques logicielles et matérielles leur permettent alors d'accéder aux contenus électroniques :

- commande d'ordinateur par le regard;
- clavier virtuel;
- souris alternatives;
- joysticks.

Déficiences mentales

En fonction des déficiences, le recours à des logiciels ou matériels adéquats est souvent nécessaire. Le logiciel "Ecrire avec des symboles 2000" (voir <http://www.unapei.org>) permet, par exemple, aux non-lecteurs d'envoyer et de recevoir des messages par Internet.

L'accessibilité doit également aider à surmonter les différences culturelles des utilisateurs (illettrisme), ainsi que certaines situations de handicap cognitif. L'ergonomie du site Internet et la stratégie de navigation ainsi que la simplification du langage sont les éléments fondamentaux de l'adaptation d'un site Internet.

Accessibilité du web

L'accessibilité du web

« L'accessibilité du Web signifie que les personnes en situation de handicap peuvent utiliser le Web. Plus précisément, qu'elles peuvent percevoir, comprendre, naviguer et interagir avec le Web, et qu'elles peuvent contribuer sur le Web. L'accessibilité du Web bénéficie aussi à d'autres, notamment les personnes âgées dont les capacités changent avec l'âge.

L'accessibilité du Web comprend tous les handicaps qui affectent l'accès au Web, ce qui inclut les handicaps visuels, auditifs, physiques, de parole, cognitifs et neurologiques. » (WAI, Web Accessibility Initiative)

Les démarches reconnues en matière d'accessibilité web

Confort de lecture: démarche volontaire basée sur le 1er niveau des WCAG + option d'affichage (publics et privés, France).

Euracert : démarche de labellisation basée sur le label Accessiweb + autres tests (publics ou privés, France).

Accessiweb: démarche de labellisation basée sur les WCAG (publics ou privés, France).

WCAG (Web Content Accessibility Guidelines): démarche volontaire, basée sur des recommandations (privés ou publics; internationale).

RGAA (Référentiel Général d'Accessibilité pour les Administrations): démarche réglementaire basée sur les WCAG (publics, France).

Confort de lecture

L'application "Confort de lecture" est transférable sur n'importe quel site Internet conforme au niveau minimum d'accessibilité recommandé par la W3C. Elle est développée par SQLI et l'association Handicapzéro afin d'apporter une réponse aux situations de handicap visuel, qu'il s'agisse de déficience légère ou de cécité totale.

Elle offre aux internautes aveugles, un accès direct optimisé aux liens et aux contenus via des plages braille ou des synthèses vocales, et aux internautes malvoyants, un affichage des pages réorganisé en fonction de paramètres qu'ils auront eux-mêmes personnalisés : couleur du fond d'écran, taille et couleur des caractères, etc.

La solution Confort de lecture s'adapte aussi bien aux sites simples que complexes ; elle est en mesure de traiter en temps réel l'intégralité des pages d'un site, qu'il soit

développé en ASP, Java, PHP ou NET. Elle peut être en mode « webservice » ou en mode « intégré » permettant une utilisation sur Internet et sur les Intranets.

Euracert

Euracert est le premier label européen de qualité pour les sites Web accessibles. Il est attribué aux sites web conformes aux recommandations WCAG du W3C/WAI. Le contrôle de conformité est réalisé à la demande des exploitants de sites web, par rapport à des documents de référence sur l'accessibilité numérique des sites. Le label Euracert est attribué après que le site web en cause ait été labellisé par l'organisme partenaire du label Euracert dans le pays de l'exploitant du site. En France, l'organisme de labellisation en matière d'accessibilité numérique, partenaire d'Euracert est l'association BrailleNet qui a créé le label AccessiWeb. La liste des sites web labellisés AccessiNet est publiée sur le site.

AccessiWeb

Le référentiel AccessiWeb a été élaboré par l'association BrailleNet et a débouché sur la création d'un label de qualité. Il s'agit d'un label privé qui ne pourra pas remplacer le RGAA. Ce label comprend 2 éléments indissociables : une liste courte de critères et une liste déployée étendue aux tests.

Les 133 critères AccessiWeb sont classés en 3 niveaux de priorité:

- Bronze (82 critères).
- Argent (20 critères supplémentaires).
- Or (31 critères supplémentaires).

Ils sont répartis dans 14 thématiques : cadres, multimédia, liens, éléments obligatoires, présentation de l'information, navigation, images, couleurs, tableaux, scripts, structuration de l'information, formulaires, consultation.

WCAG (1.0 et 2.0)

Les WCAG se composent de 14 directives (qui sont des principes généraux de conception accessible) déclinées en 65 points de contrôle (qui décrivent comment appliquer la directive).

1. Donner des alternatives textuelles complémentaires à du contenu audio et vidéo.
2. Utiliser les couleurs de manière appropriée.
3. Utiliser les balises et les feuilles de style de façon appropriée.
4. Clarifier l'utilisation du langage naturel (pour faciliter la prononciation ou l'interprétation de texte abrégé ou en langue étrangère lors de la lecture par synthèse vocale).

5. Créer des tableaux qui puissent se transformer de manière élégante.
6. Faire en sorte que les pages restent accessibles même si le visiteur n'utilise pas certaines technologies ou les a désactivées.
7. Permettre à l'utilisateur de contrôler certains éléments qui peuvent bouger, clignoter, défiler ou qui se mettent automatiquement à jour.
8. Permettre l'accès direct à des interfaces utilisateurs intégrées.
9. Créer du contenu web qui ne s'appuie pas sur un seul périphérique d'entrée ou de sortie spécifique.
10. Utiliser des solutions intermédiaires.
11. Utiliser les recommandations du W3C.
12. Donner des informations contextuelles ou d'orientation afin d'aider l'utilisateur à comprendre des pages ou des éléments complexes.
13. Utiliser des mécanismes de navigation clairs.
14. Créer des documents simples et clairs (par un langage clair, par des contenus visuels et auditifs complémentaires pour une meilleure compréhension et par une cohérence entre toutes les pages du site).

Les points de contrôle sont classés en 3 niveaux de priorité, correspondant à différents degrés d'exigence en matière d'accessibilité.

Priorité A : Permet d'assurer une accessibilité minimale. Il serait souhaitable que tout site tienne compte de ce niveau de priorité.

Priorité AA : Si cette priorité n'est pas respectée, certaines personnes handicapées pourraient ne pas accéder à toutes les parties du site.

Priorité AAA : Le plus haut niveau d'accessibilité. Il n'est pas fondamental pour un site de e-commerce (à moins de cibler prioritairement un public qui aurait certains handicaps).

Les WCAG 1.0 concernent principalement les contenus HTML et bannissent de technologies considérées comme non accessibles (pdf, flash, etc.).

Les WCAG 2.0 ont pour objectifs d'être applicables à toutes les technologies actuelles de conception de pages Web (pdf, flash, etc.). En effet, les nouvelles technologies permettent aujourd'hui aux concepteurs de les rendre lisibles par les synthèses vocales.

Les principes du WCAG 2.0:

Utilisable

Rendre toutes les fonctionnalités accessibles au clavier.

Laisser à l'utilisateur suffisamment de temps pour lire et utiliser le contenu.

Ne pas concevoir de contenu susceptible de provoquer des crises.

Fournir à l'utilisateur des éléments d'orientation pour naviguer, trouver le

contenu et se situer dans le site.

Compréhensible

Rendre le contenu textuel lisible et compréhensible.

Faire en sorte que les pages apparaissent et fonctionnent de manière prévisibles.

Aider l'utilisateur à éviter et à corriger les erreurs de saisie.

Perceptible

Proposer des équivalents textuels à tout contenu non textuel.

Proposer des versions de remplacement aux médias temporels.

Créer un contenu adaptable qui puisse être présenté de différentes manières sans perte d'information ni de structure.

Faciliter la perception visuelle et auditive du contenu.

Robuste

Optimiser la compatibilité avec les agents utilisateurs actuels et futurs y compris les technologies d'assistance.

RGAA: une obligation pour les sites web publics

Depuis l'adoption de la "Loi pour l'égalité des droits et des chances, la participation et la citoyenneté des personnes handicapées" le 11 février 2005, l'accessibilité des sites Internet publics devient une obligation légale en France. Les sites web existants devront se mettre en conformité en appliquant les règles du Référentiel Général d'Accessibilité pour les Administrations (RGAA).

Le RGAA reprend la structure des recommandations internationales, W3C/WAI/WCAG. La nouveauté repose essentiellement sur l'instauration de tests dont l'objectif est de faciliter la mise en œuvre des différents points de contrôle (chaque point de contrôle est décomposé en un ou plusieurs critères/tests, 206 critères/tests au total).

RGAA: référentiel technique

Le référentiel technique RGAA (3.0) est un document de vérification de l'accessibilité des pages Web. Il est composé des thématiques et des critères avec les tests.

Afin de faciliter la lecture et la compréhension du référentiel, la liste de contrôle synthétique (plus bas) présente seulement les critères de chaque thématique. Les mots clés des critères sont en gras pour permettre la lecture en diagonale.

Le document complet RGAA (3.0) est en annexe.

Éléments Obligatoires

Principe WCAG: Perceptible

Recommandation:

Vérifier que chaque page Web a un code source valide selon le type de document, un titre pertinent et une indication de langue par défaut. Vérifier que les balises ne sont pas utilisées uniquement pour la présentation, que les changements de langues et de direction de sens de lecture sont indiqués.

1 Critère (A)

Chaque page Web est-elle définie par **un type de document** ?

2 Critère (A)

Pour chaque page Web, le **code source est-il valide** selon le type de document spécifié (hors cas particuliers) ?

3 Critère (A)

Dans chaque page Web, **la langue par défaut** est-elle présente ?

4 Critère (A)

Pour chaque page Web ayant une langue par défaut, **le code de langue** est-il pertinent ?

5 Critère (A)

Chaque page Web a-t-elle **un titre de page** ?

6 Critère (A)

Pour chaque page Web ayant un titre de page, ce titre est-il **pertinent** ?

7 Critère (AA)

Dans chaque page Web, **chaque changement de langue** est-il indiqué dans le code source (hors cas particuliers) ?

8 Critère (AA)

Dans chaque page Web, chaque changement de langue est-il **pertinent** ?

9 Critère (A)

Dans chaque page Web, les **balises ne doivent pas** être utilisées uniquement à des fins de **présentation**. Cette règle est-elle respectée ?

10 Critère (A)

Dans chaque page Web, les **changements du sens de lecture** sont-ils signalés ?

Structuration de l'information

Principe WCAG: Perceptible

Recommandation:

Utiliser des titres, des listes, des abréviations et des citations pour structurer l'information.

1 Critère (A)

Dans chaque page Web, l'information est-elle structurée par l'**utilisation appropriée de titres** ?

2 Critère (A)

Dans chaque page Web, **la structure du document est-elle cohérente** ?

3 Critère (A)

Dans chaque page Web, chaque **liste est-elle correctement structurée** ?

4 Critère (AAA)

Dans chaque page Web, la première occurrence de **chaque abréviation** permet-elle d'en connaître la **signification** ?

5 Critère (AAA)

Dans chaque page Web, la signification de chaque abréviation est-elle **pertinente** ?

6 Critère (A)

Dans chaque page Web, chaque **citation** est-elle correctement indiquée ?

Présentation de l'information

Principe WCAG: Perceptible

Recommandation:

Utiliser des feuilles de styles pour contrôler la présentation de l'information. Vérifier l'effet de l'agrandissement des tailles de caractère sur la lisibilité. S'assurer que les liens sont correctement identifiables, que la prise de focus est signalée, que l'interlignage est suffisant et donner la possibilité à l'utilisateur de contrôler la justification des textes. S'assurer que les textes cachés sont correctement restitués et que l'information n'est pas donnée uniquement par la forme ou la position d'un élément.

1 Critère (A)

Dans le site Web, des **feuilles de styles** sont-elles utilisées pour **contrôler la présentation** de l'information ?

2 Critère (A)

Dans chaque page Web, **le contenu visible** reste-t-il présent lorsque les **feuilles de styles sont désactivées** ?

3 Critère (A)

Dans chaque page Web, l'information reste-t-elle **compréhensible** lorsque les feuilles de styles sont désactivées ?

4 Critère (AA)

Dans chaque page Web, **le texte** reste-t-il **lisible** lorsque la taille des caractères est augmentée jusqu'à 200%, au moins ?

5 Critère (AA)

Dans chaque page Web, les déclarations **CSS de couleurs de fond d'élément et de police** sont-elles correctement utilisées ?

6 Critère (A)

Dans chaque page Web, chaque **lien dont la nature n'est pas évidente est-il visible** par rapport au texte environnant ? Dans chaque page Web, chaque lien dont la nature n'est pas évidente est-il visible par rapport au texte environnant ?

7 Critère (A)

Dans chaque page Web, pour chaque élément recevant le focus, **la prise de focus** est-elle visible ?

8 Critère (AAA)

Dans chaque page Web, le choix de la **couleur de fond et de police** du texte est-il **contrôlable** par l'utilisateur ?

9 Critère (AAA)

Pour chaque page Web, le **texte ne doit pas être justifié**. Cette règle est-elle respectée ?

10 Critère (AAA)

Pour chaque page Web, en affichage **plein écran** et avec une taille de police à **200%**, chaque bloc de texte reste-t-il lisible sans l'utilisation de la barre de défilement horizontal ?

11 Critère (AAA)

Pour chaque page Web, **les blocs de texte** ont-ils une largeur inférieure ou égale à **80 caractères** (hors cas particuliers) ?

12 Critère (AAA)

Pour chaque page Web, l'**espace entre les lignes et les paragraphes** est-il suffisant ?

13 Critère (A)

Pour chaque page Web, **les textes cachés** sont-ils correctement affichés pour être **restitués** par les technologies d'assistance ?

14 Critère (A)

Dans chaque page Web, l'information ne doit pas être donnée uniquement par la forme, taille ou position. Cette règle est-elle respectée ?

15 Critère (A)

Dans chaque page Web, l'**information ne doit pas être donnée par la forme, taille ou position uniquement**. Cette règle est-elle implémentée de façon pertinente ?

Multimédia

Principe WCAG: Perceptible

Recommandation:

Images

Donner à chaque image porteuse d'information une alternative textuelle pertinente et une description détaillée si nécessaire. Remplacer les images textes par du texte stylé lorsque c'est possible.

1 Critère (A)

Chaque image a-t-elle une **alternative textuelle** ?

2 Critère (A)

Pour chaque **image de décoration** ayant une **alternative textuelle**, cette alternative est-elle **vide** ?

3 Critère (A)

Pour chaque **image porteuse d'information** ayant une alternative textuelle, cette **alternative** est-elle **pertinente** (hors cas particuliers) ?

4 Critère (A)

Pour chaque image utilisée comme **CAPTCHA** ou comme image-test, ayant une **alternative textuelle**, cette alternative permet-elle d'identifier la nature et la fonction de l'image ?

5 Critère (A)

Pour chaque image utilisée comme **CAPTCHA**, une solution d'**accès alternatif** au contenu ou à la fonction du CAPTCHA est-elle présente ?

6 Critère (A)

Chaque **image porteuse d'information** a-t-elle, si nécessaire, une **description détaillée** ?

7 Critère (A)

Pour chaque **image porteuse d'information** ayant une description détaillée, cette **description** est-elle **pertinente** ?

8 Critère (AA)

Chaque **image texte porteuse d'information**, en l'absence d'un mécanisme de remplacement, doit si possible être **remplacée par du texte stylé**. Cette règle est-elle respectée (hors cas particuliers) ?

9 Critère (AAA)

Chaque **image texte porteuse d'information**, doit si possible être **remplacée par du texte stylé**. Cette règle est-elle respectée (hors cas particuliers)?

10 Critère (A)

Chaque **légende d'image** est-elle, si nécessaire, correctement **reliée à l'image correspondante** ?

Vidéos/Audio

Principe WCAG: Perceptible/Utilisable

Recommandation: **Donner si nécessaire à chaque média temporel une transcription textuelle, des sous-titres synchronisés et une audio-description synchronisée pertinents. Donner à chaque média non temporel une alternative textuelle pertinente. Rendre possible le contrôle de la consultation de chaque média temporel et non-temporel au clavier et s'assurer de leur compatibilité avec les technologies d'assistance.**

1 Critère (A)

Chaque **média temporel** pré-enregistré a-t-il, si nécessaire, **une transcription textuelle ou une audio-description** (hors cas particuliers) ?

2 Critère (A)

Pour chaque **média temporel** pré-enregistré ayant une **transcription textuelle** ou une **audio-description synchronisée**, celles-ci sont-elles **pertinentes** (hors cas particuliers) ?

3 Critère (A)

Chaque **média temporel** synchronisé pré-enregistré a-t-il, si nécessaire, des **sous-titres synchronisés** (hors cas particuliers) ?

4 Critère (A)

Pour chaque **média temporel** synchronisé pré-enregistré ayant des **sous-titres synchronisés**, ces sous-titres sont-ils **pertinents** ?

5 Critère (AA)

Chaque **média temporel** en direct a-t-il, si nécessaire, des **sous-titres synchronisés** ou une **transcription textuelle** (hors cas particuliers) ?

6 Critère (AA) Pour chaque **média temporel** en direct ayant des **sous-titres synchronisés** ou une **transcription textuelle**, ceux-ci sont-ils pertinents ?

7 Critère (AA)

Chaque **média temporel** pré-enregistré a-t-il, si nécessaire, une **audio-description synchronisée** (hors cas particuliers) ?

8 Critère (AA)

Pour chaque **média temporel** pré-enregistré ayant une **audio-description synchronisée**, celle-ci est-elle **pertinente** ?

9 Critère (AAA)

Chaque **média temporel** pré-enregistré a-t-il, si nécessaire, une interprétation en **langue des signes** (hors cas particuliers) ?

10 Critère (AAA)

Pour chaque **média temporel** pré-enregistré ayant une interprétation en **langue des signes**, celle-ci est-elle **pertinente** ?

11 Critère (AAA)

Chaque **média temporel** pré-enregistré a-t-il, si nécessaire, une **audio-description étendue** synchronisée (hors cas particuliers) ?

12 Critère (AAA)

Pour chaque **média temporel** pré-enregistré ayant une **audio-description étendue** synchronisée, celle-ci est-elle pertinente ?

13 Critère (AAA)

Chaque **média temporel** synchronisé ou seulement vidéo a-t-il, si nécessaire, une **transcription textuelle (hors cas particuliers)** ?

14 Critère (AAA)

Pour chaque **média temporel** synchronisé ou seulement vidéo, ayant une **transcription textuelle**, celle-ci est-elle pertinente ?

15 Critère (A)

Chaque **média temporel** est-il clairement identifiable (hors **cas particuliers**) ?

16 Critère (A)

Chaque **média non temporel** a-t-il, si nécessaire, une alternative (hors **cas particuliers**) ?

17 Critère (A)

Pour chaque **média non temporel** ayant une alternative, cette alternative est-elle pertinente ?

18 Critère (A)

Chaque son déclenché automatiquement est-il **contrôlable** par l'utilisateur ?

19 Critère (AAA)

Pour chaque média temporel seulement audio pré-enregistré, les dialogues sont-ils suffisamment audibles (hors cas particuliers) ?

20 Critère (A)

La consultation de chaque **média temporel** est-elle, si nécessaire, **contrôlable par le clavier et la souris** ?

21 Critère (A)

La consultation de chaque **média non temporel** est-elle **contrôlable par le clavier et la souris** ?

22 Critère (A)

Chaque **média temporel** et **non temporel** est-il **compatible avec les technologies d'assistance** (hors **cas particuliers**) ?

Cadres

Principe WCAG: Robuste

Recommandation:

Donner à chaque cadre en ligne un titre pertinent.

1 Critère (A)

Chaque **cadre en ligne** a-t-il un **titre de cadre** ?

2 Critère (A)

Pour chaque **cadre en ligne** ayant un titre de cadre, ce **titre de cadre** est-il pertinent ?

Scripts

Principe WCAG: Perceptible

Recommandation:

Donner si nécessaire à chaque script une alternative pertinente. Rendre possible le contrôle de chaque code script au moins par le clavier et la souris et s'assurer de leur compatibilité avec les technologies d'assistance.

1 Critère (A)

Chaque **script** est-il, si nécessaire, **compatible avec les technologies d'assistance** ?

2 Critère (A)

Pour chaque **script** ayant une **alternative**, cette alternative est-elle pertinente ?

3 Critère (A)

Chaque **script** est-il **contrôlable par le clavier et la souris** (hors cas particuliers) ?

4 Critère (A)

Pour chaque **script** qui initie un **changement de contexte**, l'utilisateur est-il averti ou en a-t-il le contrôle ?

5 Critère (AAA)

Chaque **script** qui provoque une **alerte** non sollicitée est-il contrôlable par l'utilisateur (hors **cas particuliers**) ?

Couleurs

Principe WCAG: Perceptible

Recommandation:

Ne pas donner l'information uniquement par la couleur et utiliser des contrastes de couleurs suffisamment élevés.

1 Critère (A)

Dans chaque page Web, l'**information** ne doit pas être donnée uniquement par la couleur. Cette règle est-elle respectée ?

2 Critère (A)

Dans chaque page Web, l'**information** ne doit pas être donnée uniquement par la couleur. Cette règle est-elle implémentée de façon pertinente ?

3 Critère (AA)

Dans chaque page Web, le **contraste** entre la couleur du texte et la couleur de son arrière-plan est-il suffisamment élevé (hors **cas particuliers**) ?

4 Critère (A)

Dans chaque page Web, le **contraste** entre la couleur du texte et la couleur de son arrière-plan est-il amélioré (hors **cas particuliers**) ?

Tableaux

Principe WCAG: Perceptible

Recommandation:

Donner à chaque tableau de données, un résumé et un titre pertinent, 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.

1 Critère (A)

Chaque **tableau de données complexe** a-t-il un **résumé** ?

2 Critère (A)

Pour chaque **tableau de données complexe** ayant un **résumé**, celui-ci est-il pertinent ?

3 Critère (A)

Pour chaque **tableau de mise en forme**, le contenu linéarisé reste-t-il compréhensible (hors **cas particuliers**) ?

4 Critère (A)

Chaque **tableau de données** a-t-il un **titre** ?

5 Critère (A)

Pour chaque **tableau de données** ayant un **titre**, celui-ci est-il pertinent ?

6 Critère (A) Pour chaque **tableau de données**, chaque **en-tête de colonnes** et chaque **en-tête de lignes** sont-ils correctement déclarés ?

7 Critère (AA)

Pour chaque **tableau de données**, la technique appropriée permettant d'associer chaque cellule avec ses **en-têtes** est-elle utilisée ?

8 Critère (A)

Chaque **tableau de mise en forme** ne doit pas utiliser d'éléments propres aux **tableaux de données**. Cette règle est-elle respectée ?

Liens

Principe WCAG: Perceptible

Recommandation:

Donner des intitulés de lien explicites, grâce à des informations de contexte notamment, et utiliser le titre de lien le moins possible. Doubler avec des liens

ou un formulaire de navigation les zones réactive d'une image cliquable côté serveur.

1 Critère (A)

Chaque lien est-il **explicite** (hors **cas particuliers**) ? 2 Critère (A)

Pour chaque lien ayant un **titre de lien**, celui-ci est-il pertinent ?

3 Critère (AAA)

Chaque intitulé de lien seul est-il **explicite hors contexte** (hors **cas particuliers**) ?

4 Critère (A)

Pour chaque page Web, chaque **lien identique** a-t-il les mêmes fonction et destination ?

5 Critère (A)

Dans chaque page Web, chaque lien, à l'exception des **ancres**, a-t-il un **intitulé** ?

Formulaires

Principe WCAG: Perceptible

Recommandation:

Associer pour chaque formulaire chacun de ses champs à son intitulé, grouper les champs dans des blocs d'informations de même nature, structurer les listes de choix de manière pertinente, donner à chaque bouton un intitulé explicite. Vérifier la présence d'aide à la saisie, s'assurer que le contrôle de saisie est accessible et que l'utilisateur peut contrôler les données à caractère financier, juridique ou personnelles.

1 Critère (A)

Chaque **champ de formulaire** a-t-il une **étiquette** ?

2 Critère (A)

Chaque **étiquette** associée à un **champ de formulaire** est-elle pertinente ?

3 Critère (AA)

Dans chaque formulaire, chaque **étiquette** associée à un **champ de formulaire** ayant la même fonction et répété plusieurs fois dans une même page ou dans un **ensemble de pages** est-elle cohérente ?

4 Critère (A)

Dans chaque formulaire, chaque **étiquette de champ** et son **champ** associé sont-ils accolés ?

5 Critère (A)

Dans chaque formulaire, les **informations de même nature** sont-elles regroupées, si nécessaire ?

6 Critère (A)

Dans chaque formulaire, chaque regroupement de **champs de formulaire** a-t-il une légende ?

7 Critère (A)

Dans chaque formulaire, chaque légende associée à un groupement de **champs de formulaire** est-elle pertinente ?

8 Critère (A)

Dans chaque formulaire, chaque **liste de choix** est-elle structurée de manière pertinente ?

9 Critère (A)

Dans chaque formulaire, l'intitulé de chaque **bouton** est-il pertinent ?

10 Critère (A)

Dans chaque formulaire, le **contrôle de saisie** est-il utilisé de manière pertinente ?

11 Critère (AA)

Dans chaque formulaire, le **contrôle de saisie** est-il accompagné, si nécessaire, de suggestions facilitant la correction des erreurs de saisie ?

12 Critère (AA)

Pour chaque formulaire, les données à caractère **financier, juridique ou personnel** peuvent-elles être modifiées, mises à jour ou récupérées par l'utilisateur ?

13 Critère (AAA)

Pour chaque formulaire, toutes les **données** peuvent-elles être **modifiées, mises à jour ou récupérées** par l'utilisateur ?

14 Critère (AAA)

Pour chaque formulaire, des **aides à la saisie** sont-elles présentes ?

15 Critère (AAA)

Pour chaque formulaire, chaque **aide à la saisie** est-elle **pertinente** ?

Navigation

Principe WCAG: Utilisable

Recommandation:

Faciliter la navigation dans un ensemble de pages par au moins deux systèmes de navigation différents (menu de navigation, plan du site ou moteur de recherche), un fil d'Ariane et l'indication de la page active dans le menu de navigation. Identifier les groupes de liens importants et la zone de contenu et donner la possibilité de les éviter par des liens de navigation interne. S'assurer que l'ordre de tabulation est cohérent et que la page ne comporte pas de piège au clavier.

1 Critère (AA)

Chaque ensemble de pages dispose-t-il de **deux systèmes de navigation** différents, au moins (hors cas particuliers) ?

2 Critère (AA)

Dans chaque ensemble de pages, le **menu et les barres de navigation** sont-ils toujours à la **même place** (hors cas particuliers) ?

3 Critère (AA)

Dans chaque ensemble de pages, le menu et les barres de navigation ont-ils une **présentation cohérente** (hors cas particuliers) ?

4 Critère (AA)

La page "**plan du site**" est-elle **pertinente** ?

5 Critère (AA)

Dans chaque ensemble de pages, la page "**plan du site**" est-elle **atteignable de manière identique** ?

6 Critère (AA)

Dans chaque ensemble de pages, le **moteur de recherche** est-il **atteignable de manière identique** ?

7 Critère (AA)

Dans chaque page d'une **collection de pages**, des liens facilitant la navigation sont-ils présents ?

8 Critère (AAA)

Dans chaque page Web, **un fil d'Ariane** est-il **présent** (hors cas particuliers) ?

9 Critère (AAA)

Dans chaque page Web, le fil d'Ariane est-il **pertinent** ?

10 Critère (A)

Dans chaque page Web, les **groupes de liens importants** (menu, barre de navigation...) et la **zone de contenu** sont-ils **identifiés** ?

11 Critère (A)

Dans chaque page Web, des **liens d'évitement** ou d'accès rapide aux groupes de liens importants et à la zone de contenu sont-ils présents (hors cas particuliers) ?

12 Critère (AAA)

Dans chaque page Web, la **page en cours de consultation** est-elle indiquée dans le menu de navigation ?

13 Critère (A)

Dans chaque page Web, **l'ordre de tabulation** est-il cohérent ?

14 Critère (A)

Dans chaque page Web, la navigation ne doit pas contenir de **piège au clavier**. Cette règle est-elle respectée ?

Consultation

Principe WCAG: Perceptible

Recommandation:

Vérifier que l'utilisateur a le contrôle des procédés de rafraîchissement, des changements brusques de luminosité, des ouvertures de nouvelles fenêtres et des contenus en mouvement ou clignotant. Indiquer lorsqu'un contenu s'ouvre dans une nouvelle fenêtre et donner des informations relatives à la consultation des fichiers en téléchargement. Ne pas faire dépendre

l'accomplissement d'une tâche d'une limite de temps sauf si elle est essentielle et s'assurer que les données saisies sont récupérées après une interruption de session authentifiée. S'assurer que les expressions inhabituelles et le jargon sont explicités. Proposer des versions accessibles ou rendre accessibles les documents en téléchargement.

1 Critère (A)

Pour chaque page Web, l'utilisateur a-t-il le **contrôle** de chaque **limite de temps** modifiant le contenu (hors **cas particuliers**) ?

2 Critère (A)

Dans chaque page Web, pour chaque **ouverture de nouvelle fenêtre**, l'utilisateur est-il averti ?

3 Critère (A)

Dans chaque page Web, l'**ouverture d'une nouvelle fenêtre** ne doit pas être déclenchée sans **action de l'utilisateur**. Cette règle est-elle respectée ?

4 Critère (AAA)

Dans chaque page Web, **une tâche** ne doit pas requérir de **limite de temps** pour être réalisée, sauf si elle se déroule en temps réel ou si cette limite de temps est essentielle. Cette règle est-elle respectée ?

5 Critère (AAA)

Dans chaque page Web, lors d'une **interruption de session authentifiée**, les données saisies par l'utilisateur sont-elles récupérées après ré-authentification ?

6 Critère (A)

Dans chaque page Web, pour chaque **fichier en téléchargement**, des informations relatives à sa consultation sont-elles présentes (hors cas particuliers) ?

7 Critère (A)

Dans chaque page Web, chaque **document bureautique** en téléchargement possède-t-il, si nécessaire, une **version accessible** ?

8 Critère (A)

Pour chaque document bureautique ayant une **version accessible**, cette version offre-t-elle la **même information** ?

9 Critère (AAA)

Dans chaque page Web, les expressions inhabituelles, **les expressions idiomatiques** ou le jargon sont-ils explicités ?

10 Critère (AAA)

Dans chaque page Web, pour chaque expression inhabituelle ou limitée, idiomatique ou de jargon ayant une définition, cette **définition est-elle pertinente** ?

11 Critère (A)

Dans chaque page Web, chaque **contenu cryptique** (art ASCII, émoticon, syntaxe cryptique) a-t-il une **alternative** ?

12 Critère (A)

Dans chaque page Web, pour chaque **contenu cryptique** (art ASCII, émoticon, syntaxe cryptique) ayant une **alternative**, cette alternative est-elle **pertinente** ?

13 Critère (AAA)

Dans chaque page Web, pour chaque **mot dont le sens ne peut être compris** sans en connaître la prononciation, celle-ci est-elle indiquée ?

14 Critère (AAA)

Dans chaque page Web, chaque **texte** qui nécessite un **niveau de lecture** plus avancé que le premier cycle de l'enseignement secondaire a-t-il une **version alternative** ?

15 Critère (A)

Dans chaque page Web, les **changements brusques de luminosité ou les effets de flash** sont-ils correctement utilisés ?

16 Critère (AAA)

Dans chaque page Web, les **changements brusques de luminosité ou les effets de flash** ont-ils une fréquence inférieure ou égale à 3 par seconde ?

17 Critère (A)

Dans chaque page Web, chaque contenu en mouvement ou clignotant est-il **contrôlable** par l'utilisateur ?

ARIA. Construction de la sémantique

WAI-ARIA (*Web Accessibility Initiative's Accessible Rich Internet Applications*) est une spécification qui permet d'améliorer l'accessibilité des interfaces riches. Elle définit un ensemble de rôles, d'états et de propriétés applicables aux *widgets* et éléments HTML, pour que ces derniers puissent être reconnus et pleinement exploités par les utilisateurs de technologies d'assistance.

En effet, il n'existait, jusqu'à l'arrivée de HTML5, aucune balise porteuse de sens pour structurer le document. En conférant de nouvelles propriétés aux balises HTML (par le biais de l'attribut `role`), ARIA permet aux technologies d'assistance de comprendre la sémantique des différents éléments.

Les rôles ARIA

Il existe de très nombreux rôles et ceux-ci peuvent être de trois types différents:

Des rôles de structuration sémantique des documents:

directory

Pour désigner un sommaire ou une table des matières.

presentation

Contrairement aux autres, permet de supprimer toute valeur sémantique d'un élément. Le contenu d'une liste `` avec un rôle `presentation` ne sera pas restitué vocalement comme une liste.

Des rôles de « Landmarks » permettant de définir des zones de navigation:

banner

Pour désigner l'en-tête d'un site web.

complementary

Pour désigner une zone de contenu complémentaire.

main

Pour désigner la zone de contenu principale.

navigation

Pour désigner la zone de navigation.

search

Pour désigner la zone de recherche.

application

Pour contenir tout élément se comportant comme une application web.

contentinfo

Pour comprendre les données complémentaires (le pied de page du document).

form

Pour les formulaires (sauf, ceux qui sont destinés à une recherche quelconque).

Des rôles de composants d'interface:

button

Pour désigner un bouton.

dialog

Pour désigner une fenêtre modale.

progressbar

Pour désigner une barre de progression.

slider

Pour désigner une glissière.

tooltip

Pour désigner une info-bulle.

combobox

Pour désigner une liste s'affichant en dessous d'un champ texte (pour des suggestions de saisie par exemple) et qui sera reliée à une zone ayant le rôle listbox.

tablist

Pour désigner une barre d'onglets qui contiendra des tabs eux-mêmes reliés à un tabpanel.

tree

Pour désigner un arbre dépliant qui contiendra des treeitem.

Les propriétés et les états ARIA

Les propriétés et les états correspondent à des attributs comme l'attribut `id` ou `class` sur un élément HTML. Ce sont des valeurs que l'on vient appliquer à un élément ayant ou non un rôle ARIA pour lui permettre de renvoyer des informations spécifiques.

La différence entre les états et les propriétés tient au fait que la valeur d'un état va être modifiée dans le temps en fonction des actions de l'utilisateur. Par exemple, une case à cocher pourra avoir un état non coché (`aria-checked="false"`) qui sera modifié ultérieurement via javascript pour être passé à coché (`aria-checked="true"`).

Les propriétés et les états peuvent être utilisés dans quatre contextes différents.

Sur des composants d'interfaces:

aria-autocomplete

Pour indiquer si un input type text dispose d'un mécanisme d'autocomplétion.

aria-checked (état)

Pour désigner si un élément est coché.

aria-haspopup

Pour indiquer si un élément a un menu contextuel.

aria-required

Pour indiquer si un champ est un champ obligatoire.

aria-invalid (état)

Pour indiquer si un champ est en erreur.

Sur des zones « live »:

Les zones « lives » sont des parties de page dont le contenu est mis à jour via javascript alors que le focus de l'utilisateur n'est pas dans la zone en question.

aria-atomic

Pour indiquer si lors d'une mise à jour l'ensemble de la zone doit être relu ou juste la partie mise à jour.

aria-busy (état)

Pour indiquer si la zone est en cours de mise à jour.

aria-live

Pour indiquer qu'une zone sera mise à jour toute seule.

aria-relevant

Pour indiquer le type de mise à jour que le lecteur d'écran doit annoncer (les ajouts, les suppressions, les modifications de texte ou tout).

Sur des éléments pouvant être glissé/déposé:

aria-dropeffect

Pour indiquer le type d'opération qui sera effectué quand un objet sera relâché sur sa destination.

aria-grabbed (état)

Pour indiquer si un élément est en train d'être glissé.

Sur des éléments à mettre en relation:

aria-controls

Pour indiquer qu'un élément est la zone permettant le contrôle d'une autre.

aria-describedby

Pour indiquer qu'un élément est décrit par le contenu d'une autre zone.

aria-labelledby

Pour indiquer qu'un élément est titré par le contenu d'une autre zone.

Il faut toutefois noter qu'il est préférable de générer via javascript les rôles et les états amenés à être modifiés lors de l'utilisation de la page car sinon, en l'absence de javascript, cela pourrait générer des soucis pour l'utilisateur de lecteur d'écrans.

De plus, il est important de comprendre deux choses :

1. Aucune présentation visuelle ou comportement javascript n'est automatiquement ajouté par l'ajout d'ARIA.
2. ARIA ne gère pas non plus le comportement au clavier ; il sera toujours nécessaire de coder via javascript l'accès et l'utilisation au clavier des composants d'interface.

En fonction du type de composants d'interface utilisé sur le site, il est recommandé de suivre [les recommandations du W3C sur l'ergonomie de ceux-ci](#).

Recommandations pour le site de la

Le résumé des corrections à réaliser afin d'adapter le site de la aux normes RGAA 3.0 est présenté plus bas.

Eléments obligatoires

Il faut corriger les erreurs HTML et CSS signalées par les validateurs W3C : le code source (règles de syntaxe HTML, sémantique des balises, etc.)

[Voir les documents de validateur W3C en Annexe](#)

```
<!DOCTYPE html>
<html lang="fr">
<a>Lien</a>, <hX>Titre</hX>, <ul>Liste</ul>, etc.
```

Mettre un titre spécifique et pertinent à chaque page (titre hx de la page + nom du site)

```
<title>Accueil - Nom du site</title>
```

Structuration de l'information

La structure sur la page d'accueil peut être modifiée afin de faciliter la lecture et la compréhensions des informations du site. Cette page doit mettre en avant les contenus les plus forts du site. Les sections contenant du texte doivent être aérées, claires et concises.

[Voir la proposition de la nouvelle maquette en Annexe](#)

Les autre pages du site doivent avoir les gabarits identiques afin de faciliter le parcours de l'utilisateur. Sur l'ensemble du site, il est conseillé de garder le même positionnement des éléments de `header`, `aside`, `footer` et du moteur de recherche.

[Voir la proposition d'un gabarit en Annexe](#)

Sur chaque page le titre doit être renseigné avec précision. La hiérarchie cohérente des titres du contenu est à respecter de niveau 1 à 6 (<h1> premier titre; <h2> à <h6> pour titrer les sections)

Des balises sémantiques sont à ajouter pour définir les zones principales :

```
<header role="banner">
<nav role="navigation">
<main role="main">
<footer role="contentinfo">
```

Il manque “page accessibilité” et “page d’aide”.

Voir la proposition d'une nouvelle arborescence en Annexe

Présentation de l'information

Il est nécessaire de modifier la présentation du contenu (“écriture accessible”):

- conserver les accents sur les lettres capitales ;
- expliciter les abréviations à chaque première utilisation dans la page ;
- veiller à ce que les contrastes entre le texte et la couleur d’arrière-plan soient suffisants ;
- limiter la largeur des textes à 80 caractères ;
- prévoir la taille minimum de police 16px ;
- choisir la police “Web accessible” ;
- prévoir un interligne d’au moins 1,5 fois la taille du texte ;
- prévoir une marge entre les paragraphes d’au moins 1,5 fois la valeur de l’interligne ;
- prévoir un intitulé parfaitement explicite pour chaque lien.
- utiliser `em`, `rem`, `%` ou les mots clés (`small`, etc.) pour définir la taille des polices de caractères ;
- permettre l’agrandissement des caractères à 200% ;
- distinguer les liens du texte environnant.

Certains articles sont à corriger en ajoutant des éléments manquant à la présentation cohérente : “image+titre+résumé+lien”.



Multimédia

Images

Il faut ajouter l'alternative textuelle (attribut `alt`) à tous les éléments ``, `<input type="image">` et `<area>` :

`alt="Information véhiculée par l'image"` pour les images informatives;

`alt=""` pour les images décoratives ou illustratives ("" vide, sans espace);

`alt="Destination du lien"` pour les images liens;

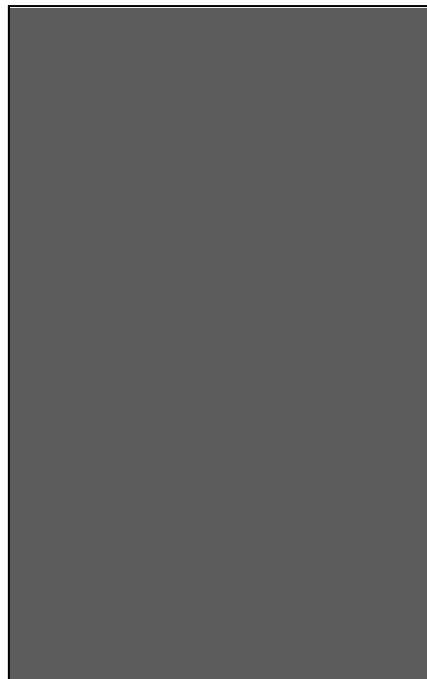
si `title` : `title = alt`.

Il est déconseillé de rédiger le texte de remplacement `alt = "texte "` en commençant par "Image..." "Lien vers..." ou "Bouton...".

Baliser les images légendées `<figure role="group">` (englobe image et sa légende) et `<figcaption>` (légende).

Toute image/icône porteuse d'information ne doit pas être insérée en CSS. Elle doit être intégrée en dur dans le code HTML (via une balise ``, via une police d'icône ou encore via une balise `<svg>` car il s'agit d'une image vectorielle).

Certaines images doivent être redimensionnées et adaptées au format web.



Son et vidéo

Les vidéos se trouvent sur la page “Galeries”. Ils ont un titre. Il faut également intégrer :

- sous-titrage / audiodescription lorsque c’est nécessaire ;
- un moyen d’accès à transcription textuelle ;
- un lecteur multimédia compatible avec l’accessibilité.



Cadres

Tout contenu, fourni par un tiers, doit être présenté dans un cadre (balise `<iframe>`). Il faut renseigner l’attribut `title` sur chaque balise `<iframe>` afin de décrire le contenu du cadre et d’indiquer sa nature, orienter l’utilisateur et lui permettre de cliquer ou non à l’intérieur.

```
<iframe src="carte.html" title="carte Nom de la carte">
```

Il est possible d’utiliser des propriétés ARIA (`aria-hidden="true"`, par exemple).

Script

Chaque script (bouton, icône, événement, etc.) doit être contrôlé à la fois à la souris et au clavier.

Il faut également que le rôle, l’état, les paramétrages et les changements de contexte soient restitués sur les éléments dynamiques. Il est nécessaire donc d’utiliser WAI-ARIA.

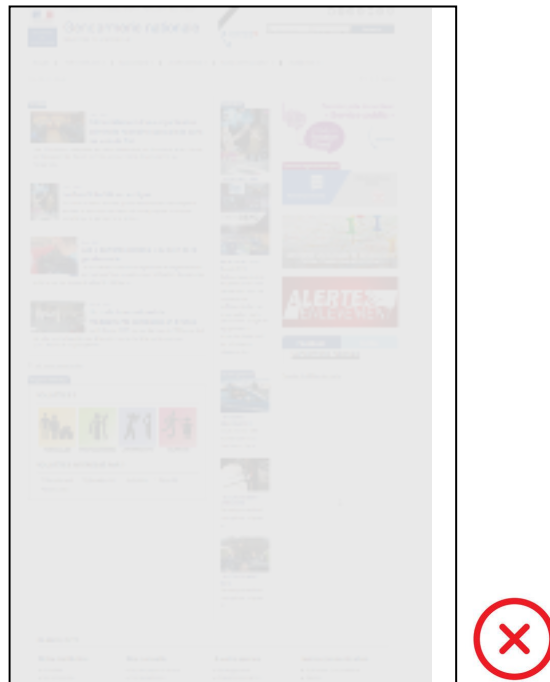
```
(aria, onclick, onchange, onmouseover, role)
```

Afin de garantir le bon ordre de changement d’élément, il faut utiliser l’attribut `tabindex="X"`.

Il est possible de spécifier ce qui doit recevoir le focus ou non et dans quel ordre :

prendre le focus : attribut `tabindex="0"`;
enlever le focus : attribut `tabindex="-1"`;

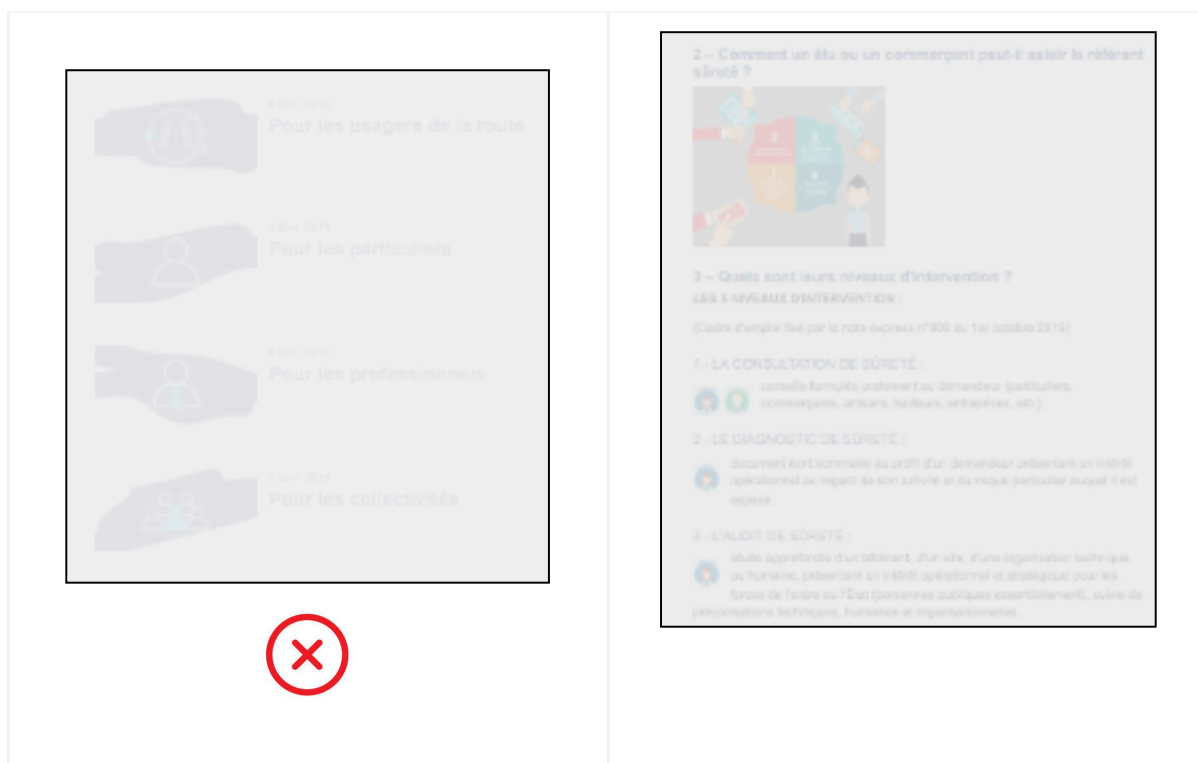
En désactivant JavaScript, certains contenus de la page d'accueil disparaissent. Il faut construire le site en pensant d'abord aux contenus, puis à la mise en page et aux effets (principe de dégradation élégante et d'amélioration progressive).



Couleurs

Les couleurs du site correspondent à la charte graphique. Certaines images informatives peuvent être améliorées en respectant ces recommandations :

- prévoir des contrastes de couleurs suffisants ;
- utiliser la couleur et la forme pour donner l'accès à une information ;
- développer un mécanisme permettant d'accentuer les contrastes.



Tableaux

Les tableaux de contenu ne sont pas présents sur le site.

Si nécessaire, des tableaux accessibles peuvent être intégrés:

- donner un titre pertinent à votre tableau avec `<caption>` ;
- ajouter un résumé à chaque tableau ;
- identifier clairement les cellules d'entête par l'utilisation de balises appropriées `<tr>`, `<td>`, `<th>`.

Pour les tableaux de données:

Utiliser un élément `<caption>` comme premier enfant de l'élément `<table>`.

Identifier les entêtes avec un élément `<th>`.

Associer les cellules avec leurs entêtes :

- soit en utilisant un attribut « `scope="col"` ou `scope="row"` » sur les `<th>`;
- soit en ajoutant un attribut « `id` » unique à chaque `<th>` et en faisant référence à cette valeur dans un attribut « `headers` » dans chaque cellule.

Pour les tableaux de mise en forme:

Permettre la lecture linéaire du tableau.

Ne pas utiliser les éléments réservés aux tableaux de données (balises ou attributs).

Ajouter un role aria : `role="presentation"` sur l'élément `<table>`.

Liens

Il est recommandé de différencier les liens et les boutons :

Les boutons `<bouton>` / `<input>` permettent à l'utilisateur de déclencher des actions.

Les liens `<a>` permettent à l'utilisateur de naviguer.

Pour rendre les liens hypertextes insérés dans votre contenu accessibles :

- donner des intitulés de liens clairs et explicites (ne pas utiliser «Lire la suite», «Cliquer ici», «En savoir plus»);
- utiliser la balise `` ou l'attribut `aria-label` ou `title` pour préciser la destination du lien ;
- ne pas utiliser l'attribut `title` si celui-ci n'est pas pertinent et n'apporte pas d'information complémentaire à l'utilisateur;
- ne pas dépasser la limite de 80 caractères.

Rendre un lien ou un bouton explicite via du texte caché consiste à :

- ajouter une balise `` dans la balise `<a>` ou `<button>` ;
- intégrer les informations permettant de rendre explicite le lien ou le bouton dans cette balise `` ;
- masquer le contenu de ce `` en CSS en le rendant invisible ou en le sortant de l'écran (hors viewport) respectivement à l'aide des propriétés CSS `"opacity: 0; ou position: absolute;"`.

Rendre un lien ou un bouton explicite via l'attribut `aria-label` consiste à :

- ajouter l'attribut `aria-label` dans la balise `<a>`, `<button>` ou `<input>`;
- renseigner cet attribut avec les informations permettant de rendre explicite le lien ou le bouton puis en reprenant à l'identique la valeur de l'intitulé du lien ou du bouton lui-même.

Rendre un lien ou un bouton explicite via l'attribut `title` consiste à :

- ajouter l'attribut `title` dans la balise `<a>`, `<button>` ou `<input />`;
- renseigner cet attribut avec les informations permettant de rendre explicite le lien ou le bouton puis en reprenant à l'identique la valeur de l'intitulé du lien ou du bouton lui-même.

Voir les exemples dans le document "Recommandations techniques"

Le lien ou bouton qui pointe vers un fichier à télécharger, les informations à renseigner sont: le nom, le format et le poids du document.

```
<a href="triptyque referent surete.pdf" title="Triptyque  
réfèrent sûreté-novembre 2013 (pdf - 96,9 kB - nouvelle  
fenêtre)" target="_blank">Document réfèrent sûreté</a>
```

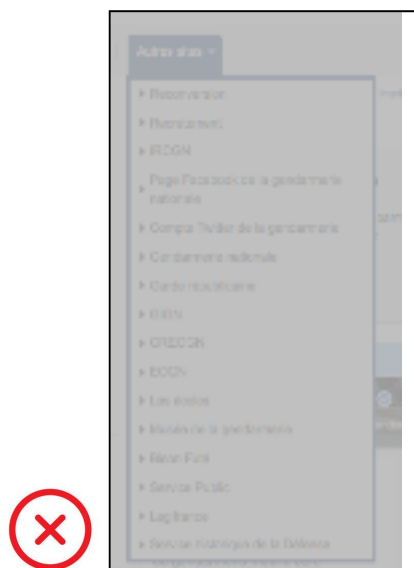


Chaque fois qu'un lien ou un bouton déclenche l'ouverture d'une nouvelle fenêtre dans le navigateur, l'utilisateur doit être prévenu par l'ajout d'une mention du type «nouvelle fenêtre»:

- soit directement dans l'intitulé ;
- soit dans l'attribut `title` ;
- soit via le texte de remplacement d'une icône/image (sous la forme d'une police d'icônes, un svg, une balise ``) ;
- soit par l'attribut `aria-label`.

Certains liens du site dirigent les utilisateurs vers d'autres sites, sans ouverture de la nouvelle fenêtre dans le navigateur. Cela doit être corrigé afin d'améliorer le parcours utilisateur.

Dans le menu de navigation, "Autres site" présente dans l'arborescence tous les liens conseillés. Ces liens ne doivent pas apparaître comme des éléments du menu de navigation.



Liens d'évitement ont pour fonction de permettre un accès rapide aux composants essentiels de la page : menu de navigation, formulaire de recherche et contenu. Les liens d'évitement renforcent l'accessibilité globale du document et permettent aux personnes naviguer au clavier, par choix ou par obligation.

```
<ul id="skip-nav">
  <li><a href="#nav">menu</a></li>
  <li><a href="#content">contenu</a></li>
  <li><a href="#search">recherche</a></li>
</ul>
```

Formulaires

Pour rendre les formulaires accessibles :

Utiliser la balise `<label>` (associée aux attributs «id» et «for») pour relier directement chaque champ de saisie au texte qui lui est associé. Associer visuellement les champs à leur intitulé.

Indiquer le caractère obligatoire dans l'étiquette. Intégrer les aides à la saisie dans les balises `<label>`.

Regrouper les champs de saisie de même nature pour une meilleure compréhension du formulaire(la balise `<fieldset>`).

Donner des titres à chaque bloc d'information en associant la balise `<fieldset>` à la balise `<legend>`. Il est aussi possible d'utiliser la balise `<optgroup>` pour grouper plusieurs choix (listes déroulantes, par exemple). Utiliser `title` ou `aria-label` pour étiqueter les champs sans intitulé visible.

Prendre en compte l'ordre de navigation en ajoutant la balise `<tabindex>` aux éléments les plus pertinents.

Alerter les utilisateurs en cas de champs invalides ou de champs non renseignés. Signaler l'erreur dans le titre de la page.

Ajouter une validation explicite des informations personnelles fournies lors d'un formulaire multiétape.

Utiliser `required` ou `aria-required="true"` sur le champ obligatoire .

Utiliser `aria-describedby` pour associer un message avec un champ.

L'utilisateur, lorsqu'il renseigne des données personnelles, financières ou juridiques, doit pouvoir, au choix :

- modifier ou annuler les données et les actions effectuées sur ces données après leur saisie ;
- vérifier et corriger les données avant la validation du formulaire ;
- confirmer explicitement, via un champ de formulaire ou une étape supplémentaire.

Les deux formulaires sont à corriger :

Identifier les différentes zones de regroupement à l'aide de la balise «`legend`» :

```
<fieldset>
  <legend>État civil</legend><br>
  Nom : <input type = "text" id = "nom" name = "nom">
  Prénom : <input type = "text" id = "prenom" name =
    "prenom">
</fieldset>
```

Rattacher les étiquettes aux champs correspondants à l'aide de la balise

« `label` » :

```
<label for = "name"> Nom</label><input type = "text" name
= "name" id = "name">
```

Formulaire de réclamation

Vous avez un problème avec nos produits ou services ? Nous sommes à votre disposition pour vous aider.

Indiquez-nous votre adresse e-mail afin que nous puissions vous contacter.

Prénom :

Nom :

Adresse e-mail :

Numéro de téléphone :

Message :

Nous écrire

Vous avez une question ou un commentaire à nous adresser ? Nous sommes à votre disposition pour vous aider.

Indiquez-nous votre adresse e-mail afin que nous puissions vous contacter.

Prénom :

Nom :

Adresse e-mail :

Numéro de téléphone :

Message :



Formulaires à corriger

Navigation

Le site propose des éléments suivants pour la navigation : menu, recherche et plan du site. Ils sont disponibles depuis le même endroit.

Des liens d'accès rapide (d'évitement) pour aller directement au contenu, à la recherche, au contact sont insérés en haut et en bas de la page.

Un fil d'Ariane est présent sur chaque page. Il indique la position courante et il permet de remonter dans l'arborescence du site. Il n'est pas obligatoire sur la page d'accueil.

Il faut ajouter un `id` unique et `tabindex=-1` sur les zones cibles.

Une page d'aide et une page d'accessibilité n'existent pas.

Voir les exemples dans le document "Recommandations techniques"

Consultation

Les animations, clignotements, changements de luminosité perturbent la lisibilité des documents en attirant l'œil du visiteur qui éprouve alors des difficultés à se concentrer sur le reste de l'information. Il convient donc de prévoir pour chaque animation un moyen pour l'internaute de l'arrêter à tout moment. Cela peut concerner les éléments suivants: gifs animés, applications flash, blocs de textes défilants et clignotants, rafraîchissement automatiques des pages.

Il est donc recommandé :

- Laisser le contrôle des rafraichissements et redirections des pages aux utilisateurs.
- Prévoir un moyen de contrôle des éléments dynamiques (pause / masquage / sans).
- Avertir lors de l'ouverture d'une nouvelles fenêtre et donner les informations sur le poids, le format et la langue des fichier en téléchargement (voir partie lien).
- Fournir des documents bureautiques accessibles ou format alternatif accessible.
- Prévoir un moyen de stopper le déclenchement de son.
- Prévoir la possibilité de supprimer / augmenter durée de session.
- Ne pas utiliser l'effet de flash.

Outils d'analyse et des tests

NVDA

<https://www.nvda-fr.org/download.php>

NonVisual desktop Access est une revue d'écran gratuite et open-source pour le système d'exploitation Microsoft Windows. En donnant des informations via une voix synthétique et le Braille, il permet aux personnes aveugles ou malvoyantes d'accéder à un ordinateur sans coût additionnel par rapport à une personne voyante.

JAWS

<http://www.accessolutions.fr/Jaws-pour-Windows.html>

Le lecteur d'écran JAWS sous Microsoft Windows, offre aux personnes déficientes visuelles un accès vocal et/ou braille du système d'exploitation et de nombreuses applications bureautiques.

Voice Over

<https://www.apple.com/fr/accessibility/mac/vision/>

VoiceOver est un logiciel de lecture d'écran développé par la société Apple et destiné aux personnes aveugles ou fortement malvoyantes. Il lit par synthèse vocale ce qui est affiché sur l'écran d'un ordinateur ou d'un appareil mobile et permet d'interagir avec ceux-ci.

Wave

<http://wave.webaim.org/>

The Wave est un outil en ligne qui scanne la page dont l'URL est entrée ; il affiche ensuite la page après analyse en ajoutant des marqueurs pour signaler les éléments qui posent problème (images non commentées, scripts java).

W3C HTML validator

<http://validator.w3.org>

CSS validator

<http://www.jigsaw.w3.org/css-validator>

HTML Tidy Clean up your Web pages

<http://www.w3.org/People/Raggett/tidy>

Firefox plugin

450 tests qualité, seo, accessibilité, performance

<https://desktop.opquast.com/fr/>

Accessibility Evaluation Toolbar

<https://addons.mozilla.org/fr/firefox/addon/accessibility-evaluation-toolb/>

Web Developer

<https://addons.mozilla.org/fr/firefox/addon/web-developer/>

aXe Developer Tools

<https://addons.mozilla.org/fr/firefox/addon/axe-devtools/>

Chrome plugin

Accessibility Developer Tools

<https://chrome.google.com/webstore/detail/accessibility-developer-t/fpkknkljclfencbdbgkenhalefipecmb>

Web Developer

<https://chrome.google.com/webstore/detail/web-developer/bfbameneiokkgbdmiekhjnmfkcnldhbm>

Color Contrast Analyzer

<https://chrome.google.com/webstore/detail/color-contrast-analyzer/dagdlcijhfbmgkjokkjjcnnfimlebccl>

Accessibility testing in Chrome Developer Tools

<https://chrome.google.com/webstore/detail/axe/lhdoppojpmngadmnindnejfpokejbdd>

Jenkins plugin

<https://github.com/Tanaguru/jenkins-tanaguru-plugin>

Outils de mesure des contrastes

<http://www.paciellogroup.com/resources/contrastanalyser/>

www.accede-web.com/notices/graphique/3-couleurs/3-1-contraste-textes-arriere-plan-alternative-contrastee

<http://contrast-finder.tanaguru.com/form.html>

<http://colorsafe.co>

<http://blog.atalan.fr/grille-contrastes-accessibilite-charte-graphique/>

Script

<http://www.w3.org/TR/wai-aria-practices/> <http://hanshillen.github.io/jqtest/>
<http://a11y.nicolas-hoffmann.net/>
<http://references.modernisation.gouv.fr/referentiel-technique-0#title-127-scripts>
http://disic.github.io/rgaa_methodologie/#script

Typographie

<https://www.smashingmagazine.com/2011/10/16-pixels-body-copy-anything-less-costly-mistake/>
<https://www.usertesting.com/blog/2014/08/06/choosing-the-right-font-a-guide-to-typography-and-user-experience/>
<http://newflux.fr/2015/10/08/les-meilleures-pratiques-de-typographie-pour-mobile/>

Terminologies

WAI Web Accessibility Initiative

WCAG Web Content Accessibility Guidelines

RGAA Référentiel Général d'Accessibilité pour les Administrations

API Application Programming Interface (interface de programmation) est un ensemble normalisé de classes, de méthodes ou de fonctions qui sert d'interface par laquelle un logiciel offre des services à d'autres logiciels.

Références

Règles pour l'accessibilité des contenus Web (WCAG) 2.0

<http://www.w3.org/Translations/WCAG20-fr>

AccessiWeb, méthode de vérification de conformité

<http://www.accessiweb.org>

RGAA – Accessibilité références modernisation.gouv.fr

<http://references.modernisation.gouv.fr/rgaa-accessibilite>

<http://wave.webaim.org>

<http://contentquality.com>

<http://www.acces-pour-tous.net>

WAI-ARIA

<http://www.w3.org/TR/wai-aria/>

<http://www.w3.org/TR/wai-aria-practices/>

https://www.w3.org/TR/wai-aria-practices/#aria_ex_widget

Confort de lecture

<http://www.confortdelecture.org/>

Euracert

<http://www.euracert.org/fr/>

Journal du net

<http://www.journaldunet.com/ebusiness/expert/51648/internet-et-handicap---les-regles-applicables-face-a-la-realite-du-net.shtml>

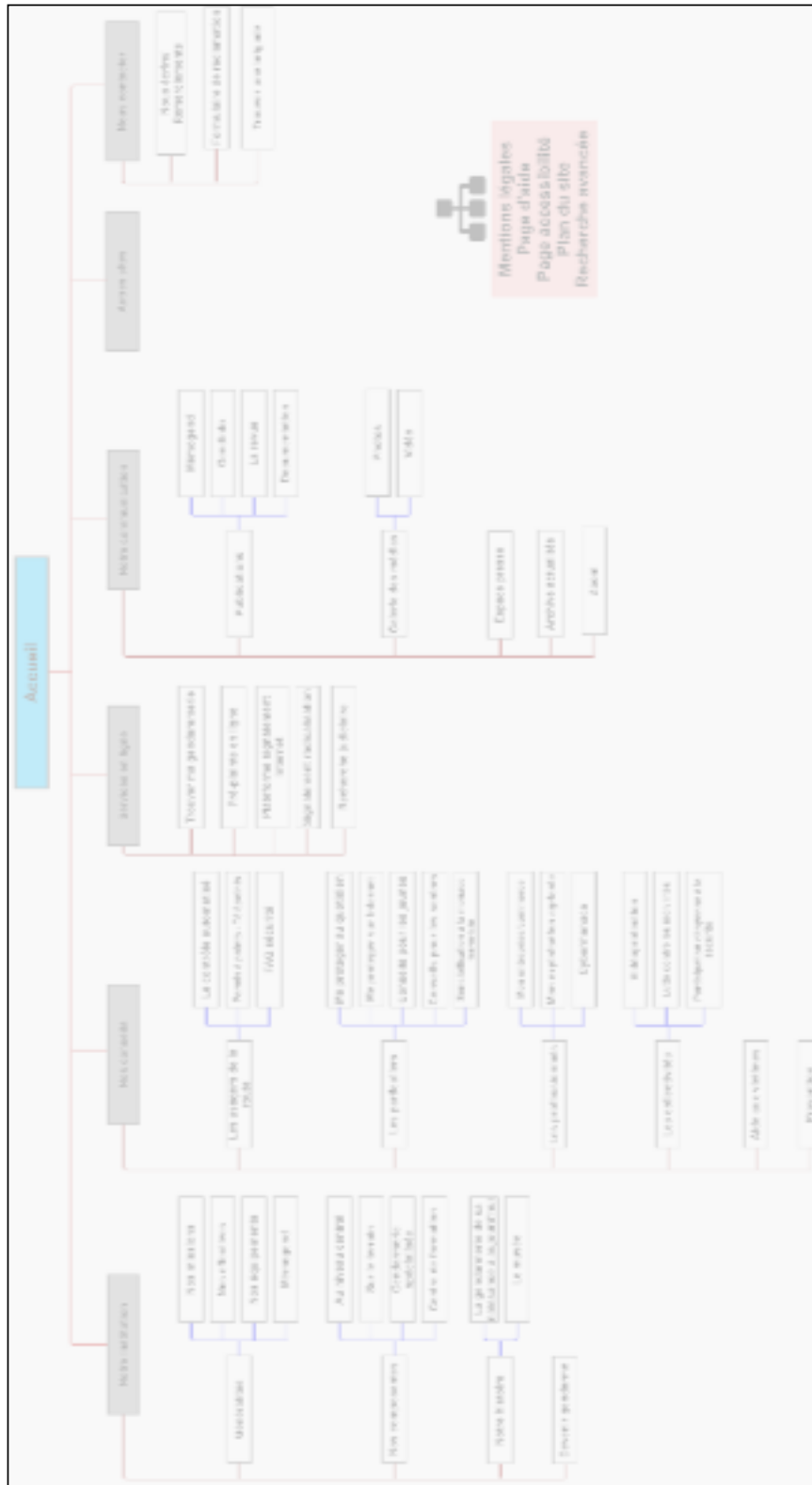
<https://msdn.microsoft.com/en-us/library/dn806236%28v=vs.85%29.aspx>

<http://references.modernisation.gouv.fr/rgaa-accessibilite/criteres.html> (version 3, 2016)

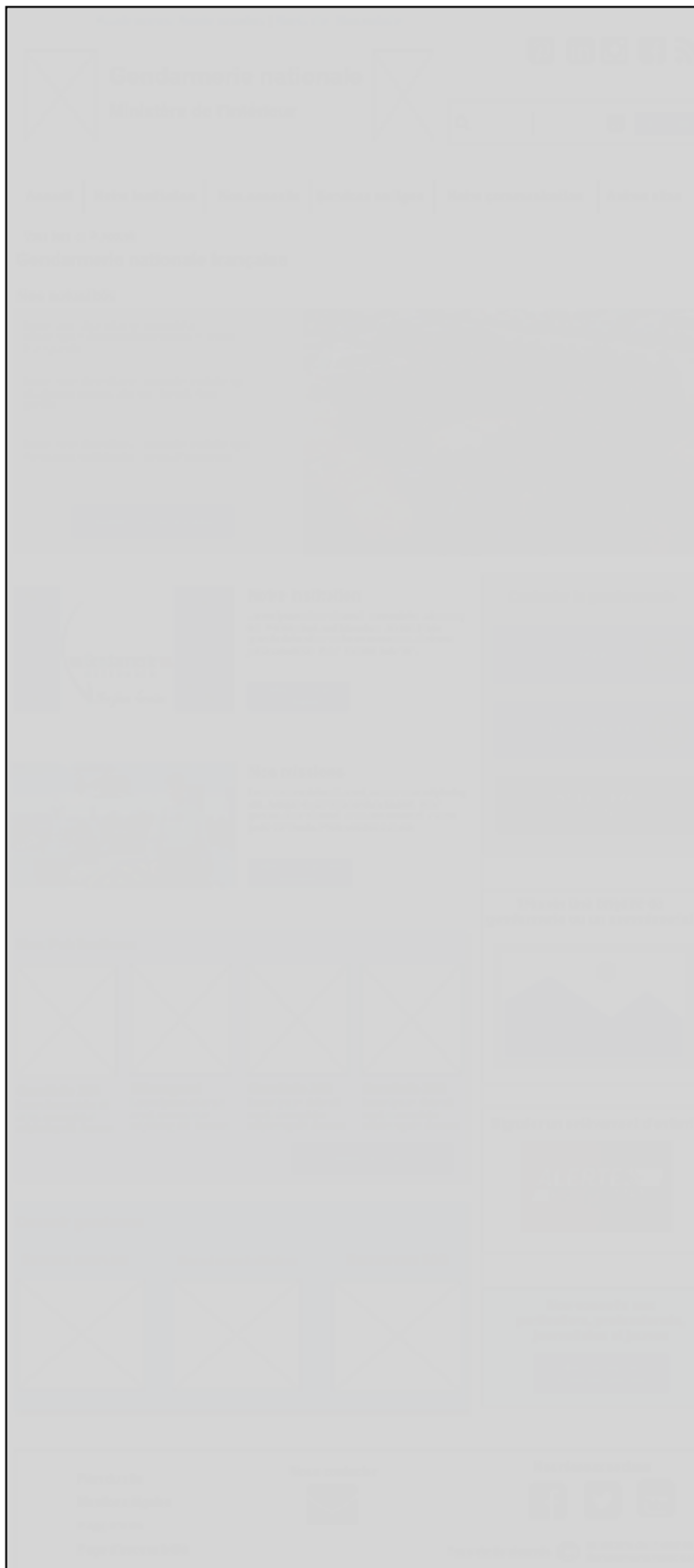
Sites accessibles

www.service-public.fr, www.loire-atlantique.fr, www.saint-maur.com,
www.ter-sncf.com

Annexe 1: Proposition Arborescence



Annexe 2: Proposition Page Accueil



Annexe 3: Proposition Gabarit des Pages

