



GOUVERNEMENT

*Liberté
Égalité
Fraternité*

**Direction
interministérielle
du numérique**

Référentiel général d'amélioration de l'accessibilité

version 4.1.2

Notes de révision

Cette édition comporte les trois premiers erratum apportés à la version 4.1 du Référentiel général d'amélioration de l'accessibilité (RGAA). Ils n'invalident pas les audits déjà réalisés.

Les corrections de cette édition concernent essentiellement les critères.

Elles corrigent le défaut suivant :

- cas particuliers manquants.

Ces corrections ont été reportées directement aux fichiers `criteres.json` et `glossaire.json` disponibles sur le dépôt GitHub du RGAA.

Les sources de cette mise à jour sont issues du dépôt RGAA.

- Critères et tests
 - 1.0.4 Ajout des cas particuliers.

1 Règles de mise en œuvre des obligations en matière d'accessibilité numérique

1.1 Notion d'accessibilité numérique

Le handicap est défini comme : « toute limitation d'activité ou restriction de participation à la vie en société subie dans son environnement par une personne en raison d'une altération substantielle, durable ou définitive d'une ou plusieurs fonctions physiques, sensorielles, mentales, cognitives ou psychiques, d'un polyhandicap ou d'un trouble de santé invalidant » (article L. 114 du code de l'action sociale et des familles).

L'accessibilité numérique consiste à rendre les services de communication au public en ligne accessibles aux personnes handicapées, c'est-à-dire :

- perceptibles : par exemple, faciliter la perception visuelle et auditive du contenu par l'utilisateur ; proposer des équivalents textuels à tout contenu non textuel ; créer un contenu qui puisse être présenté de différentes manières sans perte d'information ni de structure (par exemple avec une mise en page simplifiée) ;
- utilisables : par exemple, fournir à l'utilisateur des éléments d'orientation pour naviguer, trouver le contenu ; 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 d'épilepsie ;
- compréhensibles : par exemple, faire en sorte que les pages fonctionnent de manière prévisible ; aider l'utilisateur à corriger les erreurs de saisie.
- et robustes : par exemple, optimiser la compatibilité avec les utilisations actuelles et futures, y compris avec les technologies d'assistance.

1.2 Rappel du champ d'application

1.2.1 Contenus concernés

Comme le prévoit l'article 47 de la loi n° 2005-102 du 11 février 2005 pour l'égalité des droits et des chances, la participation et la citoyenneté des personnes handicapées, sont concernés par l'obligation d'accessibilité les services de communication au public en ligne des organismes suivants :

1. Les personnes morales de droit public ;
2. Les personnes morales de droit privé délégataires d'une mission de service public, ainsi que celles créées pour satisfaire spécifiquement des besoins d'intérêt général ayant un caractère autre qu'industriel ou commercial et dont :
 - a) Soit l'activité est financée majoritairement par une ou plusieurs personnes mentionnées aux 1° et 3° et au présent 2° ;
 - b) Soit la gestion est soumise à leur contrôle ;

- c) Soit plus de la moitié des membres de l'organe d'administration, de direction ou de surveillance sont désignés par elles ;
3. Les personnes morales de droit privé constituées par une ou plusieurs des personnes mentionnées aux 1° et 2° pour satisfaire spécifiquement des besoins d'intérêt général ayant un caractère autre qu'industriel ou commercial ;
4. Les entreprises à compter d'un seuil de chiffre d'affaires de 250 millions d'euros calculé pour chaque personne sur la base de la moyenne du chiffre d'affaires annuel réalisé en France des trois derniers exercices comptables clos antérieurement à l'année considérée.

Ne sont pas concernés les services de communication au public en ligne :

- des fournisseurs de services de médias audiovisuels ;
- des organismes de droit privé à but non lucratif qui ne fournissent ni des services essentiels pour le public, ni des services répondant spécifiquement aux besoins des personnes handicapées ou destinés à celles-ci.

Les services de communication au public en ligne sont définis comme toute mise à disposition du public ou de catégories de public, par un procédé de communication électronique, de signes, de signaux, d'écrits, d'images, de sons ou de messages de toute nature qui n'ont pas le caractère d'une correspondance privée (article 1er de la loi n° 2004-575 du 21 juin 2004 pour la confiance dans l'économie numérique). Conformément au II de l'article 47 de la loi du 11 février 2005 précitée, ils comprennent notamment :

- les sites internet, intranet, extranet ; les progiciels, dès lors qu'ils constituent des applications utilisées au travers d'un navigateur web ou d'une application mobile ;
- les applications mobiles qui sont définies comme tout logiciel d'application conçu et développé en vue d'être utilisé sur des appareils mobiles, tels que des téléphones intelligents (smartphones) et des tablettes, hors système d'exploitation ou matériel ;
- le mobilier urbain numérique, pour leur partie applicative ou interactive, hors système d'exploitation ou matériel.

1.2.2 Contenus exemptés

Certains contenus sont exemptés de l'obligation d'accessibilité et se situent hors champ de l'obligation légale :

1. Les fichiers disponibles dans des formats bureautiques publiés avant le 23 septembre 2018, sauf s'ils sont nécessaires à l'accomplissement d'une démarche administrative relevant des tâches effectuées par l'organisme concerné ;
2. Les contenus audio et vidéo préenregistrés, y compris ceux comprenant des composants interactifs, publiés avant le 23 septembre 2020 ;
3. Les contenus audio et vidéo diffusés en direct, y compris ceux comprenant des composants interactifs ;

4. Les cartes et les services de cartographie en ligne, sous réserve que, s'agissant des cartes destinées à fournir une localisation ou un itinéraire, les informations essentielles soient fournies sous une forme numérique accessible ;
5. Les contenus de tiers qui ne sont ni financés ni développés par l'organisme concerné et qui ne sont pas sous son contrôle ;
6. Les reproductions de pièces de collections patrimoniales qui ne peuvent être rendues totalement accessibles en raison :
 - a) Soit de l'incompatibilité des exigences en matière d'accessibilité avec la préservation de la pièce concernée ou l'authenticité de la reproduction notamment en termes de contraste ;
 - b) Soit de l'indisponibilité de solutions automatisées et économiques qui permettraient de transcrire facilement le texte de manuscrits ou d'autres pièces de collections patrimoniales et de le restituer sous la forme d'un contenu compatible avec l'obligation d'accessibilité ;
7. Les contenus des intranets et des extranets publiés avant le 23 septembre 2019, jusqu'à ce que ces sites fassent l'objet d'une révision en profondeur ;
8. Les contenus des sites internet et des applications mobiles qui ne sont ni nécessaires à l'accomplissement d'une démarche administrative active ni mis à jour ou modifiés après le 23 septembre 2019, notamment les archives.

Toutefois jusqu'à l'entrée en vigueur du décret n° 2019-768 du 24 juillet 2019 dans les conditions définies à son article 10, **l'État, les collectivités territoriales et les établissements publics qui en dépendent restent soumis aux obligations d'accessibilité prévues par le décret n° 2009-546 du 14 mai 2009.**

1.3 Norme de référence et niveau de conformité

Sont considérés comme conformes aux exigences légales en matière d'accessibilité les services de communication au public en ligne qui respectent la norme européenne [EN 301 549 V2.1.2 \(2018-08\)](#).

Pour les personnes mentionnées au 4° du I de l'article 47 qui font le choix de se référer aux normes internationales en matière d'accessibilité numérique, leurs contenus et services web doivent en conséquence satisfaire aux critères de succès des Règles pour l'accessibilité des contenus Web (WCAG) 2.1 de niveau simple A (A) et double A (AA).

1.4 Dérogation pour charge disproportionnée

Les exigences légales en matière d'accessibilité sont mises en œuvre dans la mesure où elles ne créent pas une charge disproportionnée pour l'organisme concerné. La charge disproportionnée est une dérogation invocable, au cas par cas, pour une fonctionnalité ou un contenu.

La charge disproportionnée peut être invoquée lorsqu'il est raisonnablement impossible à l'organisme de rendre un contenu ou une fonctionnalité accessible, notamment dans le cas où la mise en accessibilité compromettrait la capacité de l'organisme à remplir sa mission de service public ou à réaliser ses objectifs économiques.

Les contenus ou les fonctionnalités qui ne sont pas rendus accessibles en raison d'une dérogation pour charge disproportionnée sont accompagnés d'une alternative permettant à l'utilisateur d'accéder à des contenus ou fonctionnalités équivalentes, tant que la production de cette alternative ne constitue pas elle-même une charge disproportionnée.

Si le contenu ou la fonctionnalité concerne les missions principales d'un organisme chargé de mission de service public, la dérogation est obligatoirement accompagnée d'une alternative permettant d'apporter un service équivalent à l'utilisateur. Sauf exception justifiée, la dérogation ne porte pas sur l'intégralité d'un service de communication au public en ligne.

Afin de déterminer l'existence d'une telle charge, l'organisme concerné tient compte notamment des circonstances suivantes :

- la taille, les ressources et la nature de l'organisme concerné; et
- l'estimation des coûts et des avantages pour l'organisme concerné par rapport à l'avantage estimé pour les personnes handicapées, compte tenu de la fréquence et de la durée d'utilisation du service, ainsi que de l'importance du service rendu.

La taille de l'organisme est déterminée par la composition de son effectif, le nombre, la position et la répartition de ses usagers ou clients, l'importance, la diversité et le volume de ses activités et prestations, ainsi que l'étendue territoriale de ses interventions.

L'estimation des ressources tient compte :

- de son budget,
- des redevances et rémunérations perçues,
- des subventions publiques ou dons privés,
- bénéfice et recettes diverses,
- dépenses obligatoires,
- masse salariale,
- emprunts et loyers.

L'estimation des coûts peut comprendre les dépenses d'investissement, les dépenses de fonctionnement et le temps de travail, les qualifications requises. L'absence de priorité, le manque de temps ou de connaissances ne constituent pas des circonstances légitimes.

La dérogation pour charge disproportionnée n'exempte pas l'entité de produire une déclaration d'accessibilité. Les contenus et les fonctionnalités non accessibles à ce titre sont listés dans la déclaration d'accessibilité du service en ligne concerné avec la justification de la dérogation, sa durée et l'indication, le cas échéant, d'une alternative accessible.

Lorsqu'un site internet, extranet, intranet, une application mobile relève simultanément des obligations d'accessibilité imposées par l'article 47 de la loi du 11 février 2005, et de celles prescrites par les dispositions légales et réglementaires relatives à la lutte contre les discriminations et transposant la directive 2000/78/CE du 27 novembre 2000, l'appréciation de la portée et du niveau des obligations d'accessibilité, ainsi que le cas échéant de la charge disproportionnée se fonde sur les dispositions les plus favorables à l'accessibilité dans chacune de ces deux catégories de règles. Les dispositions relatives à la lutte contre les discriminations s'appliquent aux situations individuelles et

concrètes, alors que celles de l'article 47 précité, de son décret d'application et du présent référentiel sont destinées à assurer l'accessibilité des sites et applications publics à l'ensemble des usagers.

Lorsqu'une personne handicapée présente une demande d'aménagement de poste, sa demande ne peut pas lui être refusée au motif qu'elle implique un niveau d'accessibilité supérieur à ce que préconise le présent référentiel. Les exemptions et délais de mise en conformité prévus par le décret précité ne s'appliquent pas aux demandes d'aménagement de poste. La procédure à suivre et les sanctions applicables relatives aux mesures d'aménagement raisonnable sont celles prévues par les dispositions propres à la lutte contre les discriminations.

1.5 Évaluation de la conformité à la norme de référence

1.5.1 Cadre général

Afin d'évaluer la conformité du service de communication au public en ligne avec la norme de référence, l'organisme doit conduire un audit d'accessibilité. L'audit (ou évaluation) peut être effectué par l'organisme lui-même (auto-évaluation) ou par un tiers. L'évaluation est réalisée sur un échantillon de pages représentatif du service de communication au public en ligne. La vérification de la conformité des pages de l'échantillon avec les critères applicables s'effectue à l'aide des critères de contrôle du RGAA qui contiennent des tests techniques. La phase finale de l'audit est la déclaration d'accessibilité qui rend compte de la conformité des services de communication au public en ligne avec les règles applicables.

L'audit (ou évaluation) répond aux critères suivants :

- il est fiable : il revient à l'organisme concerné de veiller à la fiabilité de sa déclaration par tous moyens (recours à un prestataire externe, formation d'experts internes, audits croisés...) ;
- il est représentatif : il porte sur un échantillon représentatif des pages du service de communication au public en ligne (cf. section « Échantillon » ci-dessous).

En application de l'article 5 du décret n° 2019-768 du 24 juillet 2019 susmentionné relatif à l'accessibilité des services de communication au public en ligne aux personnes handicapées, les personnes mentionnées aux 1° à 3° de l'article 47 précité vérifient l'accessibilité de leurs services de communication au public en ligne, conformément à la norme de référence, en appliquant la méthode technique du présent référentiel. Cependant, jusqu'au 31 décembre 2019, il reste possible d'effectuer un audit de conformité tel que défini dans le RGAA3.0 mis à jour en 2017.

Les personnes mentionnées au 4° de l'article 47 précité peuvent recourir à une autre méthode de tests, à une triple condition:

- s'assurer que la méthode de test utilisée est communicable sur demande à un utilisateur ou à une administration ;
- produire une table de correspondance explicite entre les critères et tests et la norme de référence choisie ;
- l'indiquer dans la déclaration d'accessibilité.

1.5.2 Échantillon

L'échantillon sur lequel est réalisé l'audit porte au moins sur les pages suivantes, lorsqu'elles existent :

- page d'accueil (page constituant le point d'entrée principal du service de communication au public en ligne),
- page contact (page contenant les informations de contact ou le ou les formulaires permettant de contacter directement l'organisme responsable du service de communication au public en ligne),
- page mentions légales,
- page « accessibilité » (page comprenant la déclaration d'accessibilité),
- page plan du site (page récapitulant l'arborescence du site ou permettant plus largement une navigation au sein des différentes pages composant le service),
- page d'aide,
- page d'authentification.

S'ajoutent à ces pages, les pages suivantes, lorsqu'elles existent :

- au moins une page pertinente pour chaque type de service fourni et toute autre utilisation principale prévue (ex. : rubriques de 1er niveau dans l'arborescence...), y compris la fonctionnalité de recherche ;
- au moins un document téléchargeable pertinent, le cas échéant, pour chaque type de service fourni et pour toute autre utilisation principalement prévue ;
- l'ensemble des pages constituant un processus (par exemple, un formulaire de saisie ou une transaction sur plusieurs pages) ;
- des exemples de pages ayant un aspect sensiblement distinct ou présentant un type de contenu différent (ex. : page contenant des tableaux de données, des éléments multimédia, des illustrations, des formulaires, etc.).

La sélection des pages auditées ainsi que leur nombre doivent être représentatifs du service de communication au public en ligne. Le nombre de visiteurs par page peut notamment être pris en compte lors de la constitution de l'échantillon.

Enfin, s'ajoutent des pages sélectionnées au hasard représentant au moins 10 % des pages de l'échantillon décrit supra.

Sont considérées comme des pages au titre du présent référentiel les pages web et les écrans des applications mobiles.

1.5.3 Environnement de test (ou « base de référence »)

Quelques critères RGAA, notamment ceux de la thématique JavaScript, incluent des tests de restitution à effectuer sur des technologies d'assistance associées à des navigateurs et des systèmes d'exploitation.

Pour valider ces critères, il convient de définir un environnement de test (ou « base de référence »). Par défaut, il est composé des technologies d'assistance majoritairement utilisées par les personnes en situation de handicap combinées avec les navigateurs et systèmes d'exploitation pour lesquels elles sont optimisées. Cet environnement de test minimal peut être complété, le cas échéant, par des solutions libres et gratuites disponibles ou par des technologies plus anciennes, en fonction de l'usage du service de communication au public en ligne.

Lorsqu'il est possible de connaître la configuration des postes de travail, ainsi que le matériel utilisé, la base de référence est alors constituée des services réellement utilisés dans cet environnement.

L'environnement de test est présenté en 2.4.

1.5.4 Test des pages

Chaque page de l'échantillon doit être vérifiée au regard des critères qui lui sont applicables.

Il existe 3 raisons pour qu'un critère ne soit pas applicable à une page :

1. Le critère concerne un contenu ou une fonctionnalité qui n'existe pas, par exemple si la page ne comporte pas de vidéo, les critères liés aux vidéos ne seront pas applicables.
2. Le critère concerne un contenu ou un service exempté qui n'est donc pas soumis à l'obligation d'accessibilité.
3. Le critère concerne un contenu soumis à dérogation pour charge disproportionnée qui est accompagné d'une alternative numérique accessible. Par exemple, un tableau statistique avec des graphiques qui propose une alternative numérique en texte. Dans ce cas les critères applicables au contenu soumis à dérogation seront non applicables.

À noter : si le contenu soumis à dérogation pour charge disproportionnée ne propose pas d'alternative numérique accessible, les critères concernant ce contenu sont considérés comme applicables.

Les pages sont ensuite testées au regard des critères applicables. Ces tests permettent d'obtenir :

- le nombre de critères validés et non validés pour chaque page,
- le taux de conformité de chaque page.

À noter :

- un critère est validé pour une page donnée lorsque tous les éléments de la page ont passé avec succès les tests permettant de valider le critère ;
- dès lors qu'un seul élément de la page ne valide pas les tests d'un critère, le critère ne peut être validé ;
- si la page fait partie d'un processus (faire une déclaration, participer à une consultation publique, prendre un rendez-vous), un critère est validé pour une page du processus uniquement s'il est validé pour toutes les pages du processus.

1.5.5 Taux de conformité à la norme

Le taux de conformité permet de mesurer les progrès du service en ligne eu égard à la conformité aux exigences d'accessibilité.

Ce taux peut indiquer le pourcentage de critères respectés ou le niveau de conformité moyen du service en ligne.

Le pourcentage de critères RGAA respectés s'obtient en divisant le nombre de critères validés par le nombre de critères applicables.

- Critère validé : un critère est validé s'il est validé sur toutes les pages de l'échantillon. Si un critère est invalidé sur une seule page de l'échantillon, il ne pourra pas être considéré comme valide pour le calcul du taux.
- Critère applicable : pour qu'un critère soit applicable, il suffit qu'il le soit sur une seule page de l'échantillon. Ce qui a pour corollaire qu'un critère est non applicable s'il est non applicable sur toutes les pages de l'échantillon sans exception.

Le taux de moyen de conformité du service en ligne s'obtient en faisant la moyenne des taux de conformité de chaque page.

1.6 Déclaration d'accessibilité

1.6.1 Contenu de la déclaration d'accessibilité

La déclaration d'accessibilité est le résultat d'une évaluation effective de la conformité du service de communication au public en ligne à la norme de référence.

La déclaration d'accessibilité comprend :

- un état de conformité :
 - Conformité totale : si tous les critères de contrôle du RGAA sont respectés,
 - Conformité partielle : si au moins 50 % des critères de contrôle du RGAA sont respectés,
 - Non-conformité : s'il n'existe aucun résultat d'audit en cours de validité permettant de mesurer le respect des critères ou si moins de 50 % des critères de contrôle du RGAA sont respectés ;
- un signalement des contenus non accessibles, distingués selon qu'il s'agit de non-conformité avec le RGAA, de [contenus exemptés](#) ou de contenus soumis à dérogation pour [charge disproportionnée](#). Dans ce dernier cas, les dérogations doivent être expliquées et motivées. Le signalement est assorti, le cas échéant, d'une présentation des alternatives accessibles prévues ;
- des dispositifs d'assistance et de contact :
 - un mécanisme accessible (adresse électronique ou formulaire) pour permettre à toute personne de signaler à l'organisme concerné tout défaut d'accessibilité et à une personne handicapée de demander les informations correspondantes ou une solution alternative accessible ;
- la mention de la faculté pour la personne concernée de saisir le Défenseur des droits, en cas d'absence de réponse ou de solution, une fois les démarches effectuées via le mécanisme mentionné ci-dessus.

La déclaration d'accessibilité adopte obligatoirement ce format :

DÉCLARATION D'ACCESSIBILITÉ

[Nom de l'organisme concerné] s'engage à rendre [son (ses) site(s) internet, intranet, extranet et son (ses) application(s) mobile(s), etc.] accessibles conformément à l'article 47 de la loi n° 2005-102 du 11 février 2005.

À cette fin, il met en œuvre la stratégie et les actions suivantes [liens vers le schéma pluriannuel et vers le plan d'action de l'année en cours incluant le bilan des actions réalisées l'année précédente].

Cette déclaration d'accessibilité s'applique à [Nom du site web de l'application mobile, ou d'un autre service auquel la déclaration s'applique].

ÉTAT DE CONFORMITÉ

(a) [Nom du site web, de l'application mobile, ou d'un autre service concerné] est en totale conformité avec [pour les sites : le référentiel général d'amélioration de l'accessibilité / autre protocole de test utilisé ; pour les autres services de communication : protocole de test utilisé].

(b) [Nom du site web, de l'application mobile, ou d'un autre service concerné] est en conformité partielle avec [pour les sites : le référentiel général d'amélioration de l'accessibilité / autre protocole de test utilisé ; pour les autres services de communication : protocole de test utilisé] en raison [de la/des non-conformité(s)] [et/ou] [des dérogations] énumérées(s) ci-dessous.

(c) [Nom du site web de l'application mobile, ou d'un autre service concerné] n'est pas en conformité avec [pour les sites : le référentiel général d'amélioration de l'accessibilité / autre protocole de test utilisé ; pour les autres services de communication : protocole de test utilisé]. [La/les non-conformité(s)] [et/ou] [les dérogations] sont énumérées ci-dessous.

RÉSULTATS DES TESTS

L'audit de conformité réalisé [en interne / par la société NNN] révèle que :

- [pourcentage de critères RGAA respectés] des critères RGAA sont respectés.
- [Facultatif] Le taux moyen de conformité du service en ligne s'élève à [taux moyen de conformité du service en ligne].

[Détailler ici les résultats des tests ou insérer un lien vers le rapport de test]

CONTENUS NON ACCESSIBLES

Les contenus listés ci-dessous ne sont pas accessibles pour les raisons suivantes.

Non conformité

[Lister la/les non-conformité(s) du/de la/des site(s) internet/application(s) mobile(s) et/ou décrire quels section(s)/contenu(s)/fonction(s) ne sont pas encore conformes, indiquer les alternatives s'il y a lieu].

Dérogations pour charge disproportionnée

[Lister la/le/les section(s)/contenu(s)/fonction(s) non accessible(s) pour laquelle/lequel/lesquels la dérogation pour charge disproportionnée est temporairement invoquée, indiquer les alternatives s'il y a lieu].

Contenus non soumis à l'obligation d'accessibilité

[Lister la/le/les section(s)/contenu(s)/fonction(s) non accessible(s) qui n'entre(nt) pas dans le champ d'application de la législation applicable, indiquer les alternatives s'il y a lieu].

ÉTABLISSEMENT DE CETTE DÉCLARATION D'ACCESSIBILITÉ

Cette déclaration a été établie le [JJ] mois AAAA]. Elle a été mise à jour le [JJ] mois AAAA].

Technologies utilisées pour la réalisation [Du site web / De l'application mobile / Du service...]

[Liste des technologies utilisées]

Agents utilisateurs, technologies d'assistance et outils utilisés pour vérifier l'accessibilité

Les tests des pages web ont été effectués avec les combinaisons de navigateurs web et lecteurs d'écran suivants :

- [Indiquer les combinaisons utilisées pour effectuer les vérifications de conformité]

Les outils suivants ont été utilisés lors de l'évaluation :

- [Indiquer les outils utilisés]

Pages du site ayant fait l'objet de la vérification de conformité

- [Indiquer la liste des pages qui ont été testées]

RETOUR D'INFORMATION ET CONTACT

Si vous n'arrivez pas à accéder à un contenu ou à un service, vous pouvez contacter le responsable du [site internet / application mobile / Autre service] pour être orienté vers une alternative accessible ou obtenir le contenu sous une autre forme.

- Envoyer un message [url du formulaire en ligne]
- Contacter [Nom de l'entité responsable du service en ligne] [url d'une page avec les coordonnées de l'entité]

VOIES DE RECOURS

Cette procédure est à utiliser dans le cas suivant.

Vous avez signalé au responsable du site internet un défaut d'accessibilité qui vous empêche d'accéder à un contenu ou à un des services du portail et vous n'avez pas obtenu de réponse satisfaisante.

- Écrire un message au Défenseur des droits (<https://formulaire.defenseurdesdroits.fr/>)
- Contacter le délégué du Défenseur des droits dans votre région (<https://www.defenseurdesdroits.fr/saisir/delegues>)

- Envoyer un courrier par la poste (gratuit, ne pas mettre de timbre)
Défenseur des droits
Libre réponse 71120
75342 Paris CEDEX 07

1.6.2 Validité de la déclaration d'accessibilité

La déclaration d'accessibilité est valide à partir de sa date de publication. Elle doit être mise à jour :

- à la date de modification substantielle ou de refonte du site concerné.
- 3 ans après la date de publication de la déclaration, ou,
- 18 mois après la date de publication d'une nouvelle version du référentiel, pour les personnes qui appliquent la méthode technique.

Il peut cependant être souhaitable de mettre à jour plus régulièrement la déclaration d'accessibilité, y compris pour une même version de la méthode technique, afin de souligner les efforts réalisés et de mettre à jour le pourcentage de critères respectés.

1.6.3 Publication de la déclaration d'accessibilité

La déclaration d'accessibilité est publiée sur internet dans un format accessible.

Pour les sites internet, la déclaration d'accessibilité est publiée sur le site internet concerné. Elle est mise à disposition au sein d'une page accessibilité, directement accessible depuis la page d'accueil et depuis n'importe quelle page du site.

Pour les applications mobiles, elle est disponible sur le site internet de l'organisme qui a développé l'application ou apparaît avec d'autres informations disponibles lors du téléchargement de l'application. L'état de conformité est précisé au sein de l'application.

Pour les autres services de communication au public en ligne, elle est disponible sur le site internet des organismes responsables de leur gestion ou de leur mise à disposition.

La déclaration d'accessibilité fait l'objet d'un dépôt par le biais d'un téléservice selon les modalités arrêtées conjointement par le ministre chargé des personnes handicapées et le ministre chargé du numérique.

1.6.4 Réponse aux usagers

L'organisme concerné fournit en ligne aux utilisateurs la possibilité de faire des réclamations relatives à l'accessibilité de ses services de communication au public en ligne. Il accuse réception de ces réclamations conformément aux dispositions de l'article R112-3 du code des relations entre le public et l'Administration. L'accusé de réception comporte les informations mentionnées à l'article R112-5 du même code.

L'organisme concerné répond à toute réclamation dans un délai d'une semaine à compter de sa date d'envoi. Si la réclamation de l'utilisateur soulève une ou plusieurs questions complexes justifiant un délai d'examen plus long, la réponse indique un délai raisonnable pour la réponse définitive. Le caractère complexe des questions soulevées doit être dûment motivé.

L'utilisateur doit s'identifier dans sa réclamation en suivant les prescriptions de l'article R112-9-1 du même code.

Si les informations transmises par l'utilisateur dans sa réclamation sont incomplètes ou ne sont pas suffisamment claires, l'organisme concerné lui demande de les compléter ou de les clarifier et lui communique le délai raisonnable de traitement ou sa réponse définitive après réception des compléments demandés.

1.7 Schéma pluriannuel de mise en accessibilité

1.7.1 Contenu du schéma

En application du III de l'article 47 précité, le schéma pluriannuel, d'une durée maximum de trois ans, présente la politique de l'entité concernée en matière d'accessibilité numérique. A ce titre, il contient des informations sur :

- la prise en compte de l'accessibilité numérique dans la stratégie numérique de l'entité et dans sa politique en faveur de l'intégration des personnes en situation de handicap ;
- la position fonctionnelle et les missions du référent accessibilité numérique de l'entité ;
- les ressources humaines et financières affectées à l'accessibilité numérique ;
- la prise en compte des compétences ou connaissances requises dans les fiches de poste et dans les processus de recrutement ;
- les actions de formation et de sensibilisation des agents ;
- la mise en œuvre des ressources et expertises externes auxquelles il est, le cas échéant, fait appel, des moyens techniques et de l'outillage pour gérer et tester l'accessibilité numérique ;
- l'organisation interne pour mettre en œuvre les obligations d'accessibilité des services de communication au public en ligne, y compris les modalités de contrôle des services numériques et d'organisation pour le traitement des demandes des usagers ;
- l'intégration de l'accessibilité numérique dans les clauses contractuelles (appels d'offres et devis), des critères de notation et de sélection des prestataires et les procédures de recette et, le cas échéant, dans les conventions établies avec leurs opérateurs, délégataires ou partenaires.

Il présente également les travaux de mise en conformité des services de communication au public en ligne de l'entité, notamment :

- la prise en compte de l'accessibilité numérique dans les nouveaux projets ;
- la prise en compte des personnes en situation de handicap dans les tests utilisateurs ;
- les évaluations (ou audits) de conformité prévus pour l'ensemble des services de communication ;
- les mesures correctives qui seront prises pour traiter les contenus non accessibles, y compris un calendrier de mise en œuvre de ces mesures, tenant compte du caractère prioritaire des contenus les plus consultés et des services les plus utilisés ;
- les mesures d'accessibilité non obligatoires, notamment l'accès aux contenus audios et vidéos en langue des signes, la traduction de certains contenus en langage simplifié et tout autre mesure permettant de prendre en compte des critères de niveau triple AAA, des normes internationales, listés en annexe de la norme de référence ;
- le bilan des plans d'actions annuels.

Ces travaux de mise en conformité et les actions en faveur de l'accessibilité numérique sont planifiés annuellement dans des plans d'actions.

1.7.2 Publication du schéma

Le schéma pluriannuel et le plan d'action de l'année en cours sont accessibles en ligne sur le site de l'entité. Des liens vers ces documents figurent au sein de la déclaration d'accessibilité des services de communication au public en ligne dépendant de l'entité. Ils sont publiés dans un format accessible.

1.8 Mentions et pages obligatoires

1.8.1 Mention obligatoire sur la page d'accueil

La page d'accueil des services de communication au public en ligne affiche obligatoirement l'une des mentions suivantes :

- « Accessibilité : totalement conforme » si tous les critères de contrôle du RGAA sont respectés ;
- « Accessibilité : partiellement conforme » si au moins 50 % des critères de contrôle du RGAA sont respectés ;
- « Accessibilité : non conforme » s'il n'existe aucun résultat d'audit en cours de validité permettant de mesurer le respect des critères ou si moins de 50 % des critères de contrôle du RGAA sont respectés.

Cette mention peut être cliquable et conduire vers la page Accessibilité ou vers la déclaration d'accessibilité.

1.8.2 Page accessibilité et liens obligatoires

Les sites internet disposent d'une page dédiée (« accessibilité ») accessible directement depuis la page d'accueil et depuis n'importe quelle page du service de communication publique en ligne qui contient les éléments suivants :

- la déclaration d'accessibilité selon le modèle fixé par le présent référentiel ;
- le schéma pluriannuel de mise en accessibilité ou un lien vers celui-ci ;
- le plan d'actions de l'année en cours ou un lien vers celui-ci.

Il est recommandé d'utiliser une adresse internet (ou URL) standardisée telle que « www.nomdusite.extension/accessibilite ».

2 Méthode technique de vérification de la conformité à la norme de référence

2.1 Introduction

La méthode technique du RGAA permet de vérifier qu'une page web – c'est-à-dire tout contenu HTML (HTML4, XHTML1 et HTML5) - est conforme aux 50 critères de succès des niveaux A et AA de la norme internationale WCAG 2.1 qui ont été retenus dans la [norme européenne de référence](#) pour établir le niveau d'exigence légale en matière d'accessibilité numérique.

La méthode technique du RGAA propose un cadre opérationnel de vérification de la conformité aux exigences d'accessibilité.

Elle comporte 106 critères de contrôle RGAA incluant une moyenne de 2,5 tests par critères. Certains tests font référence à des techniques d'implémentation (HTML, CSS, JavaScript...) pour vérifier que le critère est respecté afin de réduire la marge d'interprétation quant au respect des normes d'accessibilité.

En cas d'absence de mise à jour du référentiel sous 3 ans pour prendre en compte de nouveaux standards ou spécifications techniques, il est possible de créer ses propres tests en complément de ceux existant.

Certains tests, concernant notamment les composants d'interface développés en JavaScript, requièrent de vérifier la restitution des contenus avec des technologies d'assistance associées à des navigateurs et des systèmes d'exploitation. L'environnement de test (ou «Base de référence») est décrit dans la partie 2.4.

À noter que la méthode technique du RGAA 4.1 ne couvre pas les applications mobiles natives, les progiciels et le mobilier urbain numérique pour lesquels il sera nécessaire de vérifier directement la mise en œuvre de la norme de référence EN 301-549 V2.1.2 notamment avec les parties :

- 5 Exigences génériques,
- 6 TIC avec communication vocale bidirectionnelle,
- 7 TIC avec capacités vidéo,
- 8 Matériels
- 11 Logiciels

(TIC = technologies de l'information et de la communication)

2.2 Critères de contrôle du RGAA

2.2.1 1. Images

| Critère 1.1 Chaque [image porteuse d'information](#) a-t-elle une [alternative textuelle](#) ?

- Test 1.1.1 : Chaque image (balise `` ou balise possédant l'attribut `WAI-ARIA role="img"`) [porteuse d'information](#) a-t-elle une [alternative textuelle](#) ?

- Test 1.1.2 : Chaque [zone](#) d'une [image réactive](#) (balise `<area>`) [porteuse d'information](#) a-t-elle une [alternative textuelle](#) ?
- Test 1.1.3 : Chaque bouton de type `image` (balise `<input>` avec l'attribut `type="image"`) a-t-il une [alternative textuelle](#) ?
- Test 1.1.4 : Chaque [zone cliquable](#) d'une image réactive côté serveur est-elle doublée d'un mécanisme utilisable quel que soit le dispositif de pointage utilisé et permettant d'accéder à la même destination ?
- Test 1.1.5 : Chaque image vectorielle (balise `<svg>`) [porteuse d'information](#), vérifie-t-elle ces conditions ?
 - La balise `<svg>` possède un attribut WAI-ARIA `role="img"` ;
 - La balise `<svg>` a une [alternative textuelle](#).
- Test 1.1.6 : Chaque [image objet](#) (balise `<object>` avec l'attribut `type="image/..."`) [porteuse d'information](#), vérifie-t-elle une de ces conditions ?
 - La balise `<object>` possède une [alternative textuelle](#) et un attribut `role="img"` ;
 - L'élément `<object>` est immédiatement suivi d'un [lien ou bouton adjacent](#) permettant d'accéder à un [contenu alternatif](#) ;
 - Un mécanisme permet à l'utilisateur de remplacer l'élément `<object>` par un [contenu alternatif](#).
- Test 1.1.7 : Chaque image embarquée (balise `<embed>` avec l'attribut `type="image/..."`) [porteuse d'information](#), vérifie-t-elle une de ces conditions ?
 - La balise `<embed>` possède une [alternative textuelle](#) et un attribut `role="img"` ;
 - L'élément `<embed>` est immédiatement suivi d'un [lien ou bouton adjacent](#) permettant d'accéder à un [contenu alternatif](#) ;
 - Un mécanisme permet à l'utilisateur de remplacer l'élément `<embed>` par un [contenu alternatif](#).
- Test 1.1.8 : Chaque image bitmap (balise `<canvas>`) [porteuse d'information](#), vérifie-t-elle une de ces conditions ?
 - La balise `<canvas>` possède une [alternative textuelle](#) et un attribut `role="img"` ;
 - Un [contenu alternatif](#) est présent entre les balises `<canvas>` et `</canvas>` ;
 - L'élément `<canvas>` est immédiatement suivi d'un [lien ou bouton adjacent](#) permettant d'accéder à un [contenu alternatif](#) ;

- Un mécanisme permet à l'utilisateur de remplacer l'élément `<canvas>` par un [contenu alternatif](#).

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F65](#) – [H24](#) – [H36](#) – [H37](#) – [H53](#)

Critère 1.2 Chaque [image de décoration](#) est-elle correctement ignorée par les technologies d'assistance ?

- Test 1.2.1 : Chaque image (balise ``) [de décoration](#), sans [légende](#), vérifie-t-elle une de ces conditions ?
 - La balise `` possède un attribut `alt` vide (`alt=""`) et est dépourvue de tout autre attribut permettant de fournir une [alternative textuelle](#) ;
 - La balise `` possède un attribut WAI-ARIA `aria-hidden="true"` ou `role="presentation"`.
- Test 1.2.2 : Chaque [zone non cliquable](#) (balise `<area>` sans attribut `href`) [de décoration](#), vérifie-t-elle une de ces conditions ?
 - La balise `<area>` possède un attribut `alt` vide (`alt=""`) et est dépourvue de tout autre attribut permettant de fournir une [alternative textuelle](#) ;
 - La balise `<area>` possède un attribut WAI-ARIA `aria-hidden="true"` ou `role="presentation"`.
- Test 1.2.3 : Chaque [image objet](#) (balise `<object>` avec l'attribut `type="image/..."`) [de décoration](#), sans [légende](#), vérifie-t-elle ces conditions ?
 - La balise `<object>` possède un attribut WAI-ARIA `aria-hidden="true"` ;
 - La balise `<object>` est dépourvue d'alternative textuelle ;
 - Il n'y a aucun texte faisant office d'alternative textuelle entre `<object>` et `</object>`.
- Test 1.2.4 : Chaque image vectorielle (balise `<svg>`) [de décoration](#), sans [légende](#), vérifie-t-elle ces conditions ?
 - La balise `<svg>` possède un attribut WAI-ARIA `aria-hidden="true"` ;
 - La balise `<svg>` et ses enfants sont dépourvus d'[alternative textuelle](#) ;
 - Les balises `<title>` et `<desc>` sont absentes ou vides ;

- La balise `<svg>` et ses enfants sont dépourvus d'attribut `title`.
- Test 1.2.5 : Chaque image bitmap (balise `<canvas>`) de décoration, sans légende, vérifie-t-elle ces conditions ?
 - La balise `<canvas>` possède un attribut WAI-ARIA `aria-hidden="true"` ;
 - La balise `<canvas>` et ses enfants sont dépourvus d'alternative textuelle ;
 - Il n'y a aucun texte faisant office d'alternative textuelle entre `<canvas>` et `</canvas>`.
- Test 1.2.6 : Chaque image embarquée (balise `<embed>` avec l'attribut `type="image/..."`) de décoration, sans légende, vérifie-t-elle ces conditions ?
 - La balise `<embed>` possède un attribut WAI-ARIA `aria-hidden="true"` ;
 - La balise `<embed>` et ses enfants sont dépourvus d'alternative textuelle.

Notes techniques

Lorsqu'une image est associée à une légende, la note technique WCAG recommande de prévoir systématiquement une alternative textuelle (cf. critère 1.9). Dans ce cas le critère 1.2 est non applicable.

Dans le cas d'une image vectorielle (balise `<svg>`) de décoration qui serait affichée au travers d'un élément `<use href="...">` enfant de l'élément `<svg>`, le test 1.2.4 s'appliquera également à l'élément `<svg>` associée par le biais de l'élément `<use>`.

Un attribut WAI-ARIA `role="presentation"` peut être utilisé sur les images de décoration et les zones non cliquables de décoration. Le rôle `"none"` introduit en ARIA 1.1 et synonyme du rôle `"presentation"` peut être aussi utilisé. Il reste préférable cependant d'utiliser le rôle `"presentation"` en attendant un support satisfaisant du rôle `"none"`.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)
- 9.4.1.2 Name, Role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA4](#) – [ARIA10](#) – [C9](#) – [F38](#) – [F39](#) – [G196](#) – [H67](#)

Critère 1.3 Pour chaque image porteuse d'information ayant une alternative textuelle, cette alternative est-elle pertinente (hors cas particuliers) ?

- Test 1.3.1 : Chaque image (balise `` ou balise possédant l'attribut WAI-ARIA `role="img"`) porteuse d'information, ayant une alternative textuelle, cette alternative est-elle pertinente (hors cas particuliers) ?
 - S'il est présent, le contenu de l'attribut `alt` est pertinent ;

- S'il est présent, le contenu de l'attribut `title` est pertinent ;
 - S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` est pertinent ;
 - S'il est présent, le [passage de texte](#) associé via l'attribut WAI-ARIA `aria-labelledby` est pertinent.
- Test 1.3.2 : Pour chaque [zone](#) (balise `<area>`) d'une [image réactive porteuse d'information](#), ayant une [alternative textuelle](#), cette alternative est-elle pertinente (hors cas particuliers) ?
 - S'il est présent, le contenu de l'attribut `alt` est pertinent ;
 - S'il est présent, le contenu de l'attribut `title` est pertinent ;
 - S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` est pertinent ;
 - S'il est présent, le [passage de texte](#) associé via l'attribut WAI-ARIA `aria-labelledby` est pertinent.
- Test 1.3.3 : Pour chaque [bouton](#) de type `image` (balise `<input>` avec l'attribut `type="image"`), ayant une [alternative textuelle](#), cette alternative est-elle pertinente (hors cas particuliers) ?
 - S'il est présent, le contenu de l'attribut `alt` est pertinent ;
 - S'il est présent, le contenu de l'attribut `title` est pertinent ;
 - S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` est pertinent ;
 - S'il est présent, le [passage de texte](#) associé via l'attribut WAI-ARIA `aria-labelledby` est pertinent.
- Test 1.3.4 : Pour chaque [image objet](#) (balise `<object>` avec l'attribut `type="image/..."`) [porteuse d'information](#), ayant une [alternative textuelle](#) ou un [contenu alternatif](#), cette alternative est-elle pertinente (hors cas particuliers) ?
 - S'il est présent, le contenu de l'attribut `title` est pertinent ;
 - S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` est pertinent ;
 - S'il est présent, le [passage de texte](#) associé via l'attribut WAI-ARIA `aria-labelledby` est pertinent ;
 - S'il est présent le [contenu alternatif](#) est pertinent.
- Test 1.3.5 : Pour chaque image embarquée (balise `<embed>` avec l'attribut `type="image/..."`) [porteuse d'information](#), ayant une [alternative textuelle](#) ou un [contenu alternatif](#), cette alternative est-elle pertinente (hors cas particuliers) ?
 - S'il est présent, le contenu de l'attribut `title` est pertinent ;

- S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` est pertinent ;
 - S'il est présent, le [passage de texte](#) associé via l'attribut WAI-ARIA `aria-labelledby` est pertinent ;
 - S'il est présent le [contenu alternatif](#) est pertinent.
- Test 1.3.6 : Pour chaque image vectorielle (balise `<svg>`) [porteuse d'information](#), ayant une [alternative textuelle](#), cette alternative est-elle pertinente (hors cas particuliers) ?
 - S'il est présent, le contenu de l'élément `<title>` est pertinent ;
 - S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` est pertinent ;
 - S'il est présent, le [passage de texte](#) associé via l'attribut WAI-ARIA `aria-labelledby` est pertinent.
 - Test 1.3.7 : Pour chaque image bitmap (balise `<canvas>`) [porteuse d'information](#), ayant une [alternative textuelle](#) ou un [contenu alternatif](#), cette alternative est-elle pertinente (hors cas particuliers) ?
 - S'il est présent, le contenu de l'attribut `title` est pertinent ;
 - S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` est pertinent ;
 - S'il est présent, le [passage de texte](#) associé via l'attribut WAI-ARIA `aria-labelledby` est pertinent ;
 - S'il est présent le [contenu alternatif](#) est pertinent.
 - Test 1.3.8 : Pour chaque image bitmap (balise `<canvas>`) [porteuse d'information](#) et ayant un [contenu alternatif](#) entre `<canvas>` et `</canvas>`, ce [contenu alternatif](#) est-il [correctement restitué par les technologies d'assistance](#) ?
 - Test 1.3.9 : Pour chaque image [porteuse d'information](#) et ayant une [alternative textuelle](#), l'[alternative textuelle](#) est-elle [courte et concise](#) (hors cas particuliers) ?

Cas particuliers

Il existe une gestion de cas particuliers lorsque l'image est utilisée comme [CAPTCHA](#) ou comme [image-test](#). Dans cette situation, où il n'est pas possible de donner une alternative pertinente sans détruire l'objet du CAPTCHA ou du test, le critère est non applicable.

Note : le cas des CAPTCHA et des images-test est traité de manière spécifique par le critère 1.4.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)

- 9.4.1.2 Name, Role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA6](#) – [ARIA9](#) – [ARIA10](#) – [F30](#) – [F71](#) – [G94](#) – [G95](#) – [G196](#)

Critère 1.4 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 ?

- Test 1.4.1 : Pour chaque image (balise ``) utilisée comme [CAPTCHA](#) ou comme [image-test](#), ayant une [alternative textuelle](#), cette alternative est-elle pertinente ?
 - S'il est présent, le contenu de l'attribut `alt` est pertinent ;
 - S'il est présent, le contenu de l'attribut `title` est pertinent ;
 - S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` est pertinent ;
 - S'il est présent, le [passage de texte](#) associé via l'attribut WAI-ARIA `aria-labelledby` est pertinent.
- Test 1.4.2 : Pour chaque zone (balise `<area>`) d'une image réactive utilisée comme [CAPTCHA](#) ou comme [image-test](#), ayant une [alternative textuelle](#), cette alternative est-elle pertinente ?
 - S'il est présent, le contenu de l'attribut `alt` est pertinent ;
 - S'il est présent, le contenu de l'attribut `title` est pertinent ;
 - S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` est pertinent ;
 - S'il est présent, le [passage de texte](#) associé via l'attribut WAI-ARIA `aria-labelledby` est pertinent.
- Test 1.4.3 : Pour chaque [bouton](#) de type image (balise `<input>` avec l'attribut `type="image"`) utilisé comme [CAPTCHA](#) ou comme [image-test](#), ayant une [alternative textuelle](#), cette alternative est-elle pertinente ?
 - S'il est présent, le contenu de l'attribut `alt` est pertinent ;
 - S'il est présent, le contenu de l'attribut `title` est pertinent ;
 - S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` est pertinent ;
 - S'il est présent, le [passage de texte](#) associé via l'attribut WAI-ARIA `aria-labelledby` est pertinent.
- Test 1.4.4 : Pour chaque [image objet](#) (balise `<object>` avec l'attribut `type="image/..."`) utilisée comme [CAPTCHA](#) ou comme [image-test](#), ayant une [alternative textuelle](#) ou un [contenu alternatif](#), cette alternative est-elle pertinente ?

- S'il est présent, le contenu de l'attribut `alt` est pertinent ;
 - S'il est présent, le contenu de l'attribut `title` est pertinent ;
 - S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` est pertinent ;
 - S'il est présent, le [passage de texte](#) associé via l'attribut WAI-ARIA `aria-labelledby` est pertinent ;
 - S'il est présent le [contenu alternatif](#) est pertinent.
- Test 1.4.5 : Pour chaque image embarquée (balise `<embed>` avec l'attribut `type="image/..."`) utilisée comme [CAPTCHA](#) ou comme [image-test](#), ayant une [alternative textuelle](#) ou un [contenu alternatif](#), cette alternative est-elle pertinente ?
 - S'il est présent, le contenu de l'attribut `alt` est pertinent ;
 - S'il est présent, le contenu de l'attribut `title` est pertinent ;
 - S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` est pertinent ;
 - S'il est présent, le [passage de texte](#) associé via l'attribut WAI-ARIA `aria-labelledby` est pertinent ;
 - S'il est présent le [contenu alternatif](#) est pertinent.
- Test 1.4.6 : Pour chaque image vectorielle (balise `<svg>`) utilisée comme [CAPTCHA](#) ou comme [image-test](#), ayant une [alternative textuelle](#), cette alternative est-elle pertinente ?
 - S'il est présent, le contenu de l'attribut `alt` est pertinent ;
 - S'il est présent, le contenu de l'attribut `title` est pertinent ;
 - S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` est pertinent ;
 - S'il est présent, le [passage de texte](#) associé via l'attribut WAI-ARIA `aria-labelledby` est pertinent.
- Test 1.4.7 : Pour chaque image bitmap (balise `<canvas>`) utilisée comme [CAPTCHA](#) ou comme [image-test](#), ayant une [alternative textuelle](#) ou un [contenu alternatif](#), cette alternative est-elle pertinente ?
 - S'il est présent, le contenu de l'attribut `alt` est pertinent ;
 - S'il est présent, le contenu de l'attribut `title` est pertinent ;
 - S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` est pertinent ;

- S'il est présent, le [passage de texte](#) associé via l'attribut WAI-ARIA `aria-labelledby` est pertinent ;
- S'il est présent le [contenu alternatif](#) est pertinent.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G100](#) – [G143](#)

Critère 1.5 Pour chaque image utilisée comme [CAPTCHA](#), une solution d'accès alternatif au contenu ou à la fonction du CAPTCHA est-elle présente ?

- Test 1.5.1 : Chaque image (balises ``, `<area>`, `<object>`, `<embed>`, `<svg>`, `<canvas>` ou possédant un attribut WAI-ARIA `role="img"`) utilisée comme [CAPTCHA](#) vérifie-t-elle une de ces conditions ?
 - Il existe une autre forme de [CAPTCHA](#) non graphique, au moins ;
 - Il existe une autre solution d'accès à la fonctionnalité qui est sécurisée par le [CAPTCHA](#).
- Test 1.5.2 : Chaque bouton associé à une image (balise `input` avec l'attribut `type="image"`) utilisée comme [CAPTCHA](#) vérifie-t-il une de ces conditions ?
 - Il existe une autre forme de [CAPTCHA](#) non graphique, au moins ;
 - Il existe une autre solution d'accès à la fonctionnalité sécurisée par le [CAPTCHA](#).

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G144](#)

Critère 1.6 Chaque image [porteuse d'information](#) a-t-elle, si nécessaire, une [description détaillée](#) ?

- Test 1.6.1 : Chaque image (balise ``) [porteuse d'information](#), qui nécessite une [description détaillée](#), vérifie-t-elle une de ces conditions ?
 - Il existe un attribut `longdesc` qui donne l'adresse (URL) d'une page ou d'un emplacement dans la page contenant la [description détaillée](#) ;
 - Il existe une [alternative textuelle](#) contenant la référence à une [description détaillée](#) adjacente à l'image ;
 - Il existe un [lien ou un bouton adjacent](#) permettant d'accéder à la [description détaillée](#).

- Test 1.6.2 : Chaque [image objet](#) (balise <object> avec l'attribut type="image/...") [porteuse d'information](#), qui nécessite une [description détaillée](#), vérifie-t-elle une de ces conditions ?
 - Il existe un attribut longdesc qui donne l'adresse (URL) d'une page ou d'un emplacement dans la page contenant la [description détaillée](#) ;
 - Il existe une [alternative textuelle](#) contenant la référence à une [description détaillée](#) adjacente à l'image ;
 - Il existe un [lien ou un bouton adjacent](#) permettant d'accéder à la [description détaillée](#).
- Test 1.6.3 : Chaque image embarquée (balise <embed>) [porteuse d'information](#), qui nécessite une [description détaillée](#), vérifie-t-elle une de ces conditions ?
 - Il existe un attribut longdesc qui donne l'adresse (URL) d'une page ou d'un emplacement dans la page contenant la [description détaillée](#) ;
 - Il existe une [alternative textuelle](#) contenant la référence à une [description détaillée](#) adjacente à l'image ;
 - Il existe un [lien ou un bouton adjacent](#) permettant d'accéder à la [description détaillée](#).
- Test 1.6.4 : Chaque [bouton](#) de type image (balise <input> avec l'attribut type="image") [porteur d'information](#), qui nécessite une [description détaillée](#), vérifie-t-il une de ces conditions ?
 - Il existe un attribut longdesc qui donne l'adresse (URL) d'une page ou d'un emplacement dans la page contenant la [description détaillée](#) ;
 - Il existe une [alternative textuelle](#) contenant la référence à une [description détaillée](#) adjacente à l'image ;
 - Il existe un [lien ou un bouton adjacent](#) permettant d'accéder à la [description détaillée](#).
- Test 1.6.5 : Chaque image vectorielle (balise <svg>) [porteuse d'information](#), qui nécessite une [description détaillée](#), vérifie-t-elle une de ces conditions ?
 - Il existe un attribut WAI-ARIA aria-label contenant l'alternative textuelle et une référence à une [description détaillée](#) adjacente ;
 - Il existe un attribut WAI-ARIA aria-labelledby associant un [passage de texte](#) faisant office d'alternative textuelle et un autre faisant office de [description détaillée](#) ;
 - Il existe un attribut WAI-ARIA aria-describedby associant un [passage de texte](#) faisant office de [description détaillée](#) ;
 - Il existe un [lien ou un bouton adjacent](#) permettant d'accéder à la [description détaillée](#).
- Test 1.6.6 : Pour chaque image vectorielle (balise <svg>) [porteuse d'information](#), ayant une [description détaillée](#), la référence éventuelle à la [description détaillée](#) dans l'attribut WAI-

ARIA `aria-label` et la [description détaillée](#) associée par l'attribut WAI-ARIA `aria-labelledby` ou `aria-describedby` sont-elles correctement restituées par les technologies d'assistance ?

- Test 1.6.7 : Chaque image bitmap (balise `<canvas>`), [porteuse d'information](#), qui nécessite une [description détaillée](#), vérifie-t-elle une de ces conditions ?
 - Il existe un attribut WAI-ARIA `aria-label` contenant l'alternative textuelle et une référence à une [description détaillée](#) adjacente ;
 - Il existe un attribut WAI-ARIA `aria-labelledby` associant un passage de texte faisant office d'alternative textuelle et un autre faisant office de [description détaillée](#) ;
 - Il existe un contenu textuel entre `<canvas>` et `</canvas>` faisant référence à une [description détaillée](#) adjacente à l'image bitmap ;
 - Il existe un contenu textuel entre `<canvas>` et `</canvas>` faisant office de [description détaillée](#) ;
 - Il existe un [lien ou bouton adjacent](#) permettant d'accéder à la [description détaillée](#).
- Test 1.6.8 : Pour chaque image bitmap (balise `<canvas>`) [porteuse d'information](#), qui implémente une référence à une [description détaillée](#) adjacente, cette référence est-elle correctement restituée par les technologies d'assistance ?
- Test 1.6.9 : Pour chaque image (balise ``, `<input>` avec l'attribut `type="image"`, `<area>`, `<object>`, `<embed>`, `<svg>`, `<canvas>`, ou possédant un attribut WAI-ARIA `role="img"`) [porteuse d'information](#), qui est accompagnée d'une [description détaillée](#) et qui utilise un attribut WAI-ARIA `aria-describedby`, l'attribut WAI-ARIA `aria-describedby` associe-t-il la [description détaillée](#) ?
- Test 1.6.10 : Chaque balise possédant un attribut WAI-ARIA `role="img"` [porteuse d'information](#), qui nécessite une [description détaillée](#), vérifie-t-elle une de ces conditions ?
 - Il existe un attribut WAI-ARIA `aria-label` contenant l'[alternative textuelle](#) et une référence à une [description détaillée](#) adjacente ;
 - Il existe un attribut WAI-ARIA `aria-labelledby` associant un [passage de texte](#) faisant office d'[alternative textuelle](#) et un autre faisant office de [description détaillée](#) ;
 - Il existe un attribut WAI-ARIA `aria-describedby` associant un [passage de texte](#) faisant office de [description détaillée](#) ;
 - Il existe un [lien ou un bouton adjacent](#) permettant d'accéder à la [description détaillée](#).

Notes techniques

Dans le cas du SVG, le manque de support de l'élément `<title>` et `<desc>` par les technologies d'assistance crée une difficulté dans le cas de l'implémentation de l'[alternative textuelle](#) de l'image

et de sa [description détaillée](#). Dans ce cas, il est recommandé d'utiliser l'attribut WAI-ARIA `aria-label` pour implémenter à la fois l'[alternative textuelle](#) courte et la référence à la [description détaillée](#) adjacente ou l'attribut WAI-ARIA `aria-labelledby` pour associer les passages de texte faisant office d'alternative courte et de [description détaillée](#).

L'utilisation de l'attribut WAI-ARIA `aria-describedby` n'est pas recommandée pour lier une image (``, `<object>`, `<embed>`, `<canvas>`) à sa [description détaillée](#), par manque de support des technologies d'assistance. Néanmoins, lorsqu'il est utilisé, l'attribut devra nécessairement faire référence à l'id de la zone contenant la [description détaillée](#).

La [description détaillée](#) adjacente peut être implémentée via une balise `<figcaption>`, dans ce cas le critère 1.9 doit être vérifié (utilisation de `<figure>` et des attributs WAI-ARIA `role="figure"` et `aria-label`, notamment).

L'attribut `longdesc` qui constitue une des conditions du test 1.6.1 (et dont la pertinence est vérifiée avec le test 1.7.1) est désormais considéré comme obsolète par la spécification HTML en cours. La vérification de cet attribut ne sera donc requise que pour les versions de la spécification HTML antérieure à HTML 5.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA6](#) – [G73](#) – [G74](#) – [G92](#) – [H45](#)

Critère 1.7 Pour chaque image [porteuse d'information](#) ayant une [description détaillée](#), cette description est-elle pertinente ?

- Test 1.7.1 : Chaque image (balise ``) [porteuse d'information](#), ayant une [description détaillée](#), vérifie-t-elle ces conditions ?
 - La [description détaillée](#) via l'adresse référencée dans l'attribut `longdesc` est pertinente ;
 - La [description détaillée](#) dans la page et signalée par l'[alternative textuelle](#) est pertinente ;
 - La [description détaillée](#) via un [lien ou un bouton adjacent](#) est pertinente ;
 - Le passage de texte associé via l'attribut WAI-ARIA `aria-describedby` est pertinent.
- Test 1.7.2 : Chaque [bouton](#) de type image (balise `<input>` avec l'attribut `type="image"`) [porteur d'information](#), ayant une [description détaillée](#), vérifie-t-il ces conditions ?
 - La [description détaillée](#) dans la page et signalée par l'[alternative textuelle](#) est pertinente ;
 - La [description détaillée](#) via un [lien ou un bouton adjacent](#) est pertinente ;
 - Le passage de texte associé via l'attribut WAI-ARIA `aria-describedby` est pertinent.

- Test 1.7.3 : Chaque [image objet](#) (balise `<object>` avec l'attribut `type="image/..."`) [porteuse d'information](#), ayant une [description détaillée](#), vérifie-t-elle ces conditions ?
 - La [description détaillée](#) dans la page et signalée par l'[alternative textuelle](#) est pertinente ;
 - La [description détaillée](#) adjacente à l'[image objet](#) est pertinente ;
 - La [description détaillée](#) via un [lien ou un bouton adjacent](#) est pertinente ;
 - Le passage de texte associé via l'attribut WAI-ARIA `aria-describedby` est pertinent.
- Test 1.7.4 : Chaque image embarquée (balise `<embed>` avec l'attribut `type="image/..."`) [porteuse d'information](#), ayant une [description détaillée](#), vérifie-t-elle ces conditions ?
 - La [description détaillée](#) dans la page et signalée par l'[alternative textuelle](#) est pertinente ;
 - La [description détaillée](#) adjacente à l'image embarquée est pertinente ;
 - La [description détaillée](#) via un [lien ou un bouton adjacent](#) est pertinente ;
 - Le passage de texte associé via l'attribut WAI-ARIA `aria-describedby` est pertinent.
- Test 1.7.5 : Chaque image vectorielle (balise `<svg>`) [porteuse d'information](#), ayant une [description détaillée](#), vérifie-t-elle ces conditions ?
 - La [description détaillée](#) dans la page et signalée par l'[alternative textuelle](#) est pertinente ;
 - La [description détaillée](#) dans la page et signalée par le texte contenu dans la balise `<desc>` ou `<title>` est pertinente ;
 - La [description détaillée](#) adjacente contenue dans la balise `<desc>` est pertinente ;
 - La [description détaillée](#) via un [lien ou un bouton adjacent](#) est pertinente ;
 - Le passage de texte associé via l'attribut WAI-ARIA `aria-describedby` est pertinent.
- Test 1.7.6 : Chaque image bitmap (balise `<canvas>`) [porteuse d'information](#), ayant une [description détaillée](#), vérifie-t-elle ces conditions ?
 - La [description détaillée](#) dans la page et signalée par l'[alternative textuelle](#) est pertinente ;
 - La [description détaillée](#) dans la page et signalée par le texte contenu entre `<canvas>` et `</canvas>` est pertinente ;
 - La [description détaillée](#) contenue entre `<canvas>` et `</canvas>` est pertinente ;
 - La [description détaillée](#) adjacente à l'image bitmap est pertinente ;
 - La [description détaillée](#) via un [lien ou un bouton adjacent](#) est pertinente ;

- Le passage de texte associé via l'attribut WAI-ARIA `aria-describedby` est pertinent.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F67](#) – [G92](#)

Critère 1.8 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) ?

- Test 1.8.1 : Chaque [image texte](#) (balise `` ou possédant un attribut WAI-ARIA `role="img"`) [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) ?
- Test 1.8.2 : Chaque bouton « [image texte](#) » (balise `<input>` avec l'attribut `type="image"`) [porteur d'information](#), en l'absence d'un [mécanisme de remplacement](#), doit si possible être remplacé par du [texte stylé](#). Cette règle est-elle respectée (hors cas particuliers) ?
- Test 1.8.3 : Chaque [image texte](#) objet (balise `<object>` avec l'attribut `type="image/..."`) [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) ?
- Test 1.8.4 : Chaque [image texte](#) embarquée (balise `<embed>` avec l'attribut `type="image/..."`) [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) ?
- Test 1.8.5 : Chaque [image texte](#) bitmap (balise `<canvas>`) [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) ?
- Test 1.8.6 : Chaque [image texte](#) SVG (balise `<svg>`) [porteuse d'information](#) et dont le texte n'est pas complètement structuré au moyen d'éléments `<text>`, 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) ?

Cas particulier

Pour ce critère, il existe une gestion de cas particulier lorsque le texte fait partie du logo, d'une dénomination commerciale, d'un [CAPTCHA](#), d'une [image-test](#) ou d'une image dont l'exactitude graphique serait considérée comme essentielle à la bonne transmission de l'information véhiculée par l'image. Dans ces situations, le critère est non applicable pour ces éléments.

Note technique

Le texte dans les images vectorielles étant du texte réel, il n'est pas concerné par ce critère.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.4.5 Images of Text (AA) / [1.4.5 Texte sous forme d'image](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [C22](#) – [C30](#) – [G136](#) – [G140](#)

Critère 1.9 Chaque [légende d'image](#) est-elle, si nécessaire, correctement reliée à l'image correspondante ?

- Test 1.9.1 : Chaque image pourvue d'une [légende](#) (balise ``, `<input>` avec l'attribut `type="image"` ou possédant un attribut WAI-ARIA `role="img"` associée à une [légende](#) adjacente), vérifie-t-elle, si nécessaire, ces conditions ?
 - L'image (balise ``, `<input>` avec l'attribut `type="image"` ou possédant un attribut WAI-ARIA `role="img"`) et sa [légende](#) adjacente sont contenues dans une balise `<figure>` ;
 - La balise `<figure>` possède un attribut WAI-ARIA `role="figure"` ou `role="group"` ;
 - La balise `<figure>` possède un attribut WAI-ARIA `aria-label` dont le contenu est identique au contenu de la [légende](#) ;
 - La [légende](#) est contenue dans une balise `<figcaption>`.
- Test 1.9.2 : Chaque [image objet](#) pourvue d'une [légende](#) (balise `<object>` avec l'attribut `type="image/..."` associée à une [légende](#) adjacente), vérifie-t-elle, si nécessaire, ces conditions ?
 - L'[image objet](#) et sa [légende](#) adjacente sont contenues dans une balise `<figure>` ;
 - La balise `<figure>` possède un attribut WAI-ARIA `role="figure"` ou `role="group"` ;
 - La balise `<figure>` possède un attribut WAI-ARIA `aria-label` dont le contenu est identique au contenu de la [légende](#) ;
 - La [légende](#) est contenue dans une balise `<figcaption>`.
- Test 1.9.3 : Chaque image embarquée pourvue d'une [légende](#) (balise `<embed>` associée à une [légende](#) adjacente), vérifie-t-elle, si nécessaire, ces conditions ?
 - L'image embarquée (balise `<embed>`) et sa [légende](#) adjacente sont contenues dans une balise `<figure>` ;
 - La balise `<figure>` possède un attribut WAI-ARIA `role="figure"` ou `role="group"` ;
 - La balise `<figure>` possède un attribut WAI-ARIA `aria-label` dont le contenu est identique au contenu de la [légende](#) ;
 - La [légende](#) est contenue dans une balise `<figcaption>`.

- Test 1.9.4 : Chaque image vectorielle pourvue d'une [légende](#) (balise `<svg>` associée à une [légende](#) adjacente), vérifie-t-elle, si nécessaire, ces conditions ?
 - L'image vectorielle (balise `<svg>`) et sa [légende](#) adjacente sont contenues dans une balise `<figure>` ;
 - La balise `<figure>` possède un attribut WAI-ARIA `role="figure"` ou `role="group"` ;
 - La balise `<figure>` possède un attribut WAI-ARIA `aria-label` dont le contenu est identique au contenu de la [légende](#) ;
 - La [légende](#) est contenue dans une balise `<figcaption>`.
- Test 1.9.5 : Chaque image bitmap pourvue d'une [légende](#) (balise `<canvas>` associée à une [légende](#) adjacente), vérifie-t-elle, si nécessaire, ces conditions ?
 - L'image bitmap (balise `<canvas>`) et sa [légende](#) adjacente sont contenues dans une balise `<figure>` ;
 - La balise `<figure>` possède un attribut WAI-ARIA `role="figure"` ou `role="group"` ;
 - La balise `<figure>` possède un attribut WAI-ARIA `aria-label` dont le contenu est identique au contenu de la [légende](#) ;
 - La [légende](#) est contenue dans une balise `<figcaption>`.

Notes techniques

L'implémentation d'un attribut WAI-ARIA `role="group"` ou `role="figure"` sur l'élément parent `<figure>` est destiné à pallier le manque de support actuel des éléments `<figure>` par les technologies d'assistance. L'utilisation d'un élément `<figcaption>` pour associer une [légende](#) à une image impose au minimum l'utilisation d'un attribut WAI-ARIA `aria-label` sur l'élément parent `<figure>` dont le contenu sera identique au contenu de l'élément `<figcaption>`. Pour s'assurer d'un support optimal, il peut également être fait une association explicite entre le contenu de l'[alternative textuelle](#) de l'image et le contenu de l'élément `<figcaption>`, par exemple :

```
<figcaption>Photo : crédit  
xxx</figcaption>
```

Les attributs WAI-ARIA `aria-labelledby` et `aria-describedby` ne peuvent pas être utilisés actuellement par manque de support par les technologies d'assistance.

Note : les images légendées doivent par ailleurs respecter le critère 1.1 et le critère 1.3 relatifs aux images porteuses d'information.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)

- 9.4.1.2 Name, Role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA4](#) – [ARIA6](#) – [G140](#)

2.2.2 2. Cadres

Critère 2.1 Chaque cadre a-t-il un titre de cadre ?

- Test 2.1.1 : Chaque cadre (balise `<iframe>` ou `<frame>`) a-t-il un attribut `title` ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.4.1.2 Name, Role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [H64](#)

Critère 2.2 Pour chaque cadre ayant un titre de cadre, ce titre de cadre est-il pertinent ?

- Test 2.2.1 : Pour chaque cadre (balise `<iframe>` ou `<frame>`) ayant un attribut `title`, le contenu de cet attribut est-il pertinent ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.4.1.2 Name, Role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [H64](#)

2.2.3 3. Couleurs

Critère 3.1 Dans chaque page web, l'information ne doit pas être donnée uniquement par la couleur. Cette règle est-elle respectée ?

- Test 3.1.1 : Pour chaque mot ou ensemble de mots dont la mise en couleur est porteuse d'information, l'information ne doit pas être donnée uniquement par la couleur. Cette règle est-elle respectée ?
- Test 3.1.2 : Pour chaque indication de couleur donnée par un texte, l'information ne doit pas être donnée uniquement par la couleur. Cette règle est-elle respectée ?
- Test 3.1.3 : Pour chaque image véhiculant une information, l'information ne doit pas être donnée uniquement par la couleur. Cette règle est-elle respectée ?
- Test 3.1.4 : Pour chaque propriété CSS déterminant une couleur et véhiculant une information, l'information ne doit pas être donnée uniquement par la couleur. Cette règle est-elle respectée ?
- Test 3.1.5 : Pour chaque média temporel véhiculant une information, l'information ne doit pas être donnée uniquement par la couleur. Cette règle est-elle respectée ?

- Test 3.1.6 : Pour chaque [média non temporel véhiculant une information](#), l'[information](#) ne doit pas être donnée uniquement par la couleur. Cette règle est-elle respectée ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)
- 9.1.4.1 Use of color (A) / [1.4.1 Utilisation de la couleur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G14](#) – [G111](#) – [G117](#) – [G138](#) – [G182](#) – [G205](#)

Critère 3.2 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) ?

- Test 3.2.1 : Dans chaque page web, le texte et le texte en image sans effet de graisse d'une taille restituée inférieure à 24px vérifient-ils une de ces conditions (hors cas particuliers) ?
 - Le rapport de [contraste](#) entre le texte et son arrière-plan est de 4.5:1, au moins ;
 - Un mécanisme permet à l'utilisateur d'afficher le texte avec un rapport de [contraste](#) de 4.5:1, au moins.
- Test 3.2.2 : Dans chaque page web, le texte et le texte en image en gras d'une taille restituée inférieure à 18,5px vérifient-ils une de ces conditions (hors cas particuliers) ?
 - Le rapport de [contraste](#) entre le texte et son arrière-plan est de 4.5:1, au moins ;
 - Un mécanisme permet à l'utilisateur d'afficher le texte avec un rapport de [contraste](#) de 4.5:1, au moins.
- Test 3.2.3 : Dans chaque page web, le texte et le texte en image sans effet de graisse d'une taille restituée supérieure ou égale à 24px vérifient-ils une de ces conditions (hors cas particuliers) ?
 - Le rapport de [contraste](#) entre le texte et son arrière-plan est de 3:1, au moins ;
 - Un mécanisme permet à l'utilisateur d'afficher le texte avec un rapport de [contraste](#) de 3:1, au moins.
- Test 3.2.4 : Dans chaque page web, le texte et le texte en image en gras d'une taille restituée supérieure ou égale à 18,5px vérifient-ils une de ces conditions (hors cas particuliers) ?
 - Le rapport de [contraste](#) entre le texte et son arrière-plan est de 3:1, au moins ;
 - Un mécanisme permet à l'utilisateur d'afficher le texte avec un rapport de [contraste](#) de 3:1, au moins.
- Test 3.2.5 : Dans le [mécanisme qui permet d'afficher un rapport de contraste](#) conforme, le rapport de contraste entre le texte et la couleur d'arrière-plan est-il suffisamment élevé ?

Cas particuliers

Dans ces situations, les critères sont non applicables pour ces éléments :

- Le texte fait partie d'un logo ou d'un nom de marque d'un organisme ou d'une société ;
- Le texte ou l'image de texte est purement décoratif ;
- Le texte fait partie d'une image véhiculant une information mais le texte lui-même n'apporte aucune information essentielle ;
- Le texte ou l'image de texte fait partie d'un élément d'interface sur lequel aucune action n'est possible (par exemple un bouton avec l'attribut `disabled`).

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.4.3 Contrast (Minimum) (AA) / [1.4.3 Contraste \(minimum\)](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [C29](#) – [G18](#) – [G136](#) – [G145](#) – [G148](#) – [G174](#)

Critère 3.3 Dans chaque page web, les couleurs utilisées dans les [composants d'interface](#) ou les éléments graphiques porteurs d'informations sont-elles suffisamment contrastées (hors cas particuliers) ?

- Test 3.3.1 : Dans chaque page web, le rapport de [contraste](#) entre les couleurs d'un [composants d'interface](#) dans ses différents états et la [couleur d'arrière-plan contiguë](#) vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - Le rapport de [contraste](#) est de 3:1, au moins ;
 - Un [mécanisme](#) permet un rapport de [contraste](#) de 3:1, au moins.
- Test 3.3.2 : Dans chaque page web, le rapport de [contraste](#) des différentes couleurs composant un [élément graphique](#), lorsqu'elles sont nécessaires à sa compréhension, et la [couleur d'arrière-plan contiguë](#), vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - Le rapport de [contraste](#) est de 3:1, au moins ;
 - Un [mécanisme](#) permet un rapport de [contraste](#) de 3:1, au moins.
- Test 3.3.3 : Dans chaque page web, le rapport de [contraste](#) des différentes [couleurs contiguës](#) entre elles d'un [élément graphique](#), lorsqu'elles sont nécessaires à sa compréhension, vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - Le rapport de [contraste](#) est de 3:1, au moins ;
 - Un [mécanisme](#) permet un rapport de [contraste](#) de 3:1, au moins.

- Test 3.3.4 : Dans le [mécanisme qui permet d'afficher un rapport de contraste](#) conforme, les couleurs du composant ou des éléments graphiques porteurs d'informations qui le composent, sont-elles suffisamment contrastées ?

Cas particuliers

Les cas suivants sont non applicables pour ce critère :

- Composant d'interface inactif (par exemple, un bouton avec un attribut `disabled`) sur lequel aucune action n'est possible ;
- Composant d'interface pour lequel l'apparence est gérée par les styles natifs du navigateur sans aucune modification par l'auteur (par exemple, le style au focus natif dans Chrome ou Firefox) ;
- Composant d'interface pour lequel la couleur n'est pas nécessaire pour identifier le composant ou son état (par exemple, un groupe de liens faisant office de navigation dont la position dans la page, la taille et la couleur du texte permettent de comprendre qu'il s'agit de liens même si la couleur du soulignement des liens avec le fond blanc n'a pas un ratio de 3:1 et que le texte lui a un ratio de 4.5:1) ;
- [Élément graphique](#) ou parties d'élément graphique non porteur d'information ou ayant une alternative (description longue, informations identiques visibles dans la page) ;
- [Élément graphique](#) ou parties d'élément graphique faisant partie d'un logo ou du nom de marque d'un organisme ou d'une société ;
- [Élément graphique](#) ou parties d'élément graphique dont la présentation est essentielle à l'information véhiculée (par exemple, drapeaux, logotypes, photos de personnes ou de scènes, captures d'écran, diagrammes médicaux, carte de chaleurs) ;
- [Élément graphique](#) ou parties d'élément graphique dynamiques dont le contraste au survol / focus est suffisant.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.4.11 Non-text Contrast (AA) / [1.4.11 Contraste du contenu non textuel](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F78](#) – [G18](#) – [G145](#) – [G174](#) – [G183](#) – [G195](#) – [G207](#)

2.2.4 4. Multimédia

Critère 4.1 Chaque [média temporel](#) pré-enregistré a-t-il, si nécessaire, une [transcription textuelle](#) ou une [audiodescription](#) (hors cas particuliers) ?

- Test 4.1.1 : Chaque [média temporel](#) pré-enregistré seulement audio, vérifie-t-il, si nécessaire, l'une de ces conditions (hors cas particuliers) ?

- Il existe une [transcription textuelle](#) accessible via un [lien ou bouton adjacent](#) ;
- Il existe une [transcription textuelle](#) adjacente clairement identifiable.
- Test 4.1.2 : Chaque [média temporel](#) pré-enregistré seulement vidéo vérifie-t-il, si nécessaire, l'une de ces conditions (hors cas particuliers) ?
 - Il existe une [version alternative « audio seulement »](#) accessible via un [lien ou bouton adjacent](#) ;
 - Il existe une [version alternative « audio seulement »](#) adjacente clairement identifiable ;
 - Il existe une [transcription textuelle](#) accessible via un [lien ou bouton adjacent](#) ;
 - Il existe une [transcription textuelle](#) adjacente clairement identifiable ;
 - Il existe une [audiodescription](#) synchronisée ;
 - Il existe une version alternative avec une [audiodescription](#) synchronisée accessible via un [lien ou bouton adjacent](#).
- Test 4.1.3 : Chaque [média temporel](#) synchronisé pré-enregistré vérifie-t-il, si nécessaire, une de ces conditions (hors cas particuliers) ?
 - Il existe une [transcription textuelle](#) accessible via un [lien ou bouton adjacent](#) ;
 - Il existe une [transcription textuelle](#) adjacente clairement identifiable ;
 - Il existe une [audiodescription](#) synchronisée ;
 - Il existe une version alternative avec une [audiodescription](#) synchronisée accessible via un [lien ou bouton adjacent](#).

Cas particuliers

Il existe une gestion de cas particulier lorsque :

- Le [média temporel](#) est utilisé à des fins décoratives (c'est-à-dire qu'il n'apporte aucune information) ;
- Le [média temporel](#) est lui-même une alternative à un contenu de la page (une vidéo en langue des signes ou la vocalisation d'un texte, par exemple) ;
- Le [média temporel](#) est utilisé pour accéder à une version agrandie ;
- Le [média temporel](#) est utilisé comme un [CAPTCHA](#) ;
- Le [média temporel](#) fait partie d'un test qui deviendrait inutile si la [transcription textuelle](#), les [sous-titres synchronisés](#) ou l'[audiodescription](#) étaient communiqués ;

- Pour les services de l'État, les collectivités territoriales et leurs établissements : si le [média temporel](#) a été publié entre le 23 septembre 2019 et le 23 septembre 2020 sur un site internet, intranet ou extranet créé depuis le 23 septembre 2018, il est exempté de l'obligation d'accessibilité ;
- Pour les personnes de droit privé mentionnées aux 2° à 4° du I de l'article 47 de la loi du 11 février 2005 : si le [média temporel](#) a été publié avant le 23 septembre 2020, il est exempté de l'obligation d'accessibilité.

Dans ces situations, le critère est non applicable.

Ce cas particulier s'applique également aux critères 4.2, 4.3, 4.5.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.2.1 Audio-only and Video-only (Prerecorded) (A) / [1.2.1 Contenus seulement audio et seulement vidéo \(pré-enregistrés\)](#)
- 9.1.2.3 Audio Description or Media Alternative (Prerecorded) (A) / [1.2.3 Audio-description ou version de remplacement pour un média temporel \(pré-enregistré\)](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G8](#) – [G58](#) – [G69](#) – [G78](#) – [G158](#) – [G159](#) – [G166](#) – [G173](#) – [H96](#) – [SM6](#) – [SM7](#)

Critère 4.2 Pour chaque [média temporel](#) pré-enregistré ayant une [transcription textuelle](#) ou une [audiodescription](#) synchronisée, celles-ci sont-elles pertinentes (hors cas particuliers) ?

- Test 4.2.1 : Pour chaque [média temporel](#) pré-enregistré seulement audio, ayant une [transcription textuelle](#), celle-ci est-elle pertinente (hors cas particuliers) ?
- Test 4.2.2 : Chaque [média temporel](#) pré-enregistré seulement vidéo vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - La [transcription textuelle](#) est pertinente ;
 - L'[audiodescription](#) synchronisée est pertinente ;
 - L'[audiodescription](#) synchronisée de la version alternative est pertinente ;
 - La version alternative audio seulement est pertinente.
- Test 4.2.3 : Chaque [média temporel](#) synchronisé pré-enregistré vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - La [transcription textuelle](#) est pertinente ;
 - L'[audiodescription](#) synchronisée est pertinente ;
 - L'[audiodescription](#) synchronisée de la version alternative est pertinente.

Cas particulier

Voir cas particuliers critère 4.1.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.2.1 Audio-only and Video-only (Prerecorded) (A) / [1.2.1 Contenus seulement audio et seulement vidéo \(pré-enregistrés\)](#)
- 9.1.2.3 Audio Description or Media Alternative (Prerecorded) (A) / [1.2.3 Audio-description ou version de remplacement pour un média temporel \(pré-enregistré\)](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F30](#) – [F67](#) – [SM6](#) – [SM7](#)

Critère 4.3 Chaque [média temporel](#) synchronisé pré-enregistré a-t-il, si nécessaire, des [sous-titres synchronisés](#) (hors cas particuliers) ?

- Test 4.3.1 : Chaque [média temporel](#) synchronisé pré-enregistré vérifie-t-il, si nécessaire, l'une de ces conditions (hors cas particuliers) ?
 - Le [média temporel](#) synchronisé possède des [sous-titres synchronisés](#) ;
 - Il existe une version alternative possédant des [sous-titres synchronisés](#) accessible via un [lien](#) ou bouton adjacent.
- Test 4.3.2 : Pour chaque [média temporel](#) synchronisé pré-enregistré possédant des [sous-titres synchronisés](#) diffusés via une balise <track>, la balise <track> possède-t-elle un attribut `kind="captions"` ?

Cas particulier

Voir cas particuliers critère 4.1.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.2.2 Captions (Prerecorded) (A) / [1.2.2 Sous-titres \(pré-enregistrés\)](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F74](#) – [F75](#) – [G58](#) – [G87](#) – [G93](#) – [H95](#) – [SM11](#) – [SM12](#)

Critère 4.4 Pour chaque [média temporel](#) synchronisé pré-enregistré ayant des [sous-titres synchronisés](#), ces sous-titres sont-ils pertinents ?

- Test 4.4.1 : Pour chaque [média temporel](#) synchronisé pré-enregistré ayant des [sous-titres synchronisés](#), ces sous-titres sont-ils pertinents ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.2.2 Captions (Prerecorded) (A) / [1.2.2 Sous-titres \(pré-enregistrés\)](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F8](#) – [F74](#) – [F75](#) – [G87](#) – [G93](#) – [SM11](#) – [SM12](#)

Critère 4.5 Chaque [média temporel](#) pré-enregistré a-t-il, si nécessaire, une [audiodescription synchronisée](#) (hors cas particuliers) ?

- Test 4.5.1 : Chaque [média temporel](#) pré-enregistré seulement vidéo vérifie-t-il, si nécessaire, une de ces conditions (hors cas particuliers) ?
 - Il existe une [audiodescription](#) synchronisée ;
 - Il existe une version alternative avec une [audiodescription](#) synchronisée.
- Test 4.5.2 : Chaque [média temporel](#) synchronisé pré-enregistré vérifie-t-il, si nécessaire, une de ces conditions (hors cas particuliers) ?
 - Il existe une [audiodescription](#) synchronisée ;
 - Il existe une version alternative avec une [audiodescription](#) synchronisée.

Cas particulier

Voir cas particuliers critère 4.1.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.2.5 Audio Description (Prerecorded) (AA) / [1.2.5 Audio-description \(pré-enregistrée\)](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G8](#) – [G58](#) – [G78](#) – [G173](#) – [H96](#) – [SM1](#) – [SM2](#) – [SM6](#) – [SM7](#)

Critère 4.6 Pour chaque [média temporel](#) pré-enregistré ayant une [audiodescription synchronisée](#), celle-ci est-elle pertinente ?

- Test 4.6.1 : Pour chaque [média temporel](#) pré-enregistré seulement vidéo ayant une [audiodescription](#) synchronisée, celle-ci est-elle pertinente ?
- Test 4.6.2 : Pour chaque [média temporel](#) synchronisé ayant une [audiodescription](#) synchronisée, celle-ci est-elle pertinente ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.2.5 Audio Description (Prerecorded) (AA) / [1.2.5 Audio-description \(pré-enregistrée\)](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [SM1](#) – [SM2](#) – [SM6](#) – [SM7](#)

Critère 4.7 Chaque [média temporel](#) est-il clairement identifiable (hors cas particuliers) ?

- Test 4.7.1 : Pour chaque [média temporel](#) seulement son, seulement vidéo ou synchronisé, le contenu textuel adjacent permet-il d'identifier clairement le [média temporel](#) (hors cas particuliers) ?

Cas particulier

Il existe une gestion de cas particulier lorsque le [média temporel](#) est utilisé à des fins décoratives (c'est-à-dire qu'il n'apporte aucune information). Dans cette situation, le critère est non applicable.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G68](#) – [G100](#)

Critère 4.8 Chaque [média non temporel](#) a-t-il, si nécessaire, une alternative (hors cas particuliers) ?

- Test 4.8.1 : Chaque [média non temporel](#) vérifie-t-il, si nécessaire, une de ces conditions (hors cas particuliers) ?
 - Un [lien ou un bouton adjacent](#), clairement identifiable, permet d'accéder à une page contenant une alternative ;
 - Un [lien ou un bouton adjacent](#), clairement identifiable, permet d'accéder à une alternative dans la page.
- Test 4.8.2 : Chaque [média non temporel](#) associé à une alternative vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - La page référencée par le [lien ou bouton adjacent](#) est accessible ;
 - L'alternative dans la page, référencée par le [lien ou bouton adjacent](#), est accessible.

Cas particuliers

Il existe une gestion de cas particulier lorsque :

- Le [média non temporel](#) est utilisé à des fins décoratives (c'est-à-dire qu'il n'apporte aucune information) ;
- Le [média non temporel](#) est diffusé dans un [environnement maîtrisé](#) ;
- Le [média non temporel](#) est inséré via JavaScript en vérifiant la présence et la version du plug-in, en remplacement d'un [contenu alternatif](#) déjà présent.

Dans ces situations, le critère est non applicable.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [H35](#) – [H46](#)

Critère 4.9 Pour chaque [média non temporel](#) ayant une alternative, cette alternative est-elle pertinente ?

- Test 4.9.1 : Pour chaque [média non temporel](#) ayant une alternative, cette alternative permet-elle d'accéder au même contenu et à des fonctionnalités similaires ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F30](#) – [H46](#)

Critère 4.10 Chaque son déclenché automatiquement est-il [contrôlable](#) par l'utilisateur ?

- Test 4.10.1 : Chaque séquence sonore déclenchée automatiquement via une balise `<object>`, `<video>`, `<audio>`, `<embed>`, `<bgsound>` ou un code JavaScript vérifie-t-elle une de ces conditions ?
 - La séquence sonore a une durée inférieure ou égale à 3 secondes ;
 - La séquence sonore peut être stoppée sur action de l'utilisateur ;
 - Le volume de la séquence sonore peut être contrôlé par l'utilisateur indépendamment du contrôle de volume du système.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.4.2 Audio Control (A) / [1.4.2 Contrôle du son](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F23](#) – [F93](#) – [G60](#) – [G170](#) – [G171](#)

Critère 4.11 La consultation de chaque [média temporel](#) est-elle, si nécessaire, [contrôlable par le clavier et tout dispositif de pointage](#) ?

- Test 4.11.1 : Chaque [média temporel](#) a-t-il, si nécessaire, les fonctionnalités de contrôle de sa consultation ?
- Test 4.11.2 : Pour chaque [média temporel](#), chaque fonctionnalité vérifie-t-elle une de ces conditions ?
 - La fonctionnalité est [accessible par le clavier et tout dispositif de pointage](#) ;

- Une fonctionnalité [accessible par le clavier et tout dispositif de pointage](#) permettant de réaliser la même action est présente dans la page.
- Test 4.11.3 : Pour chaque [média temporel](#), chaque fonctionnalité vérifie-t-elle une de ces conditions ?
 - La fonctionnalité est [activable par le clavier et tout dispositif de pointage](#) ;
 - Une fonctionnalité [activable par le clavier et tout dispositif de pointage](#) permettant de réaliser la même action est présente dans la page.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.1.1 Keyboard (A) / [2.1.1 Clavier](#)
- 9.2.1.2 No Keyboard Trap (A) / [2.1.2 Pas de piège au clavier](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G4](#) – [G90](#) – [G202](#)

Critère 4.12 La consultation de chaque [média non temporel](#) est-elle [contrôlable par le clavier et tout dispositif de pointage](#) ?

- Test 4.12.1 : Pour chaque [média non temporel](#), chaque fonctionnalité vérifie-t-elle une de ces conditions ?
 - La fonctionnalité est [accessible par le clavier et tout dispositif de pointage](#) ;
 - Une fonctionnalité [accessible par le clavier et tout dispositif de pointage](#) permettant de réaliser la même action est présente dans la page.
- Test 4.12.2 : Pour chaque [média non temporel](#), chaque fonctionnalité vérifie-t-elle une de ces conditions ?
 - La fonctionnalité est [activable par le clavier et tout dispositif de pointage](#) ;
 - Une fonctionnalité [activable par le clavier et tout dispositif de pointage](#) permettant de réaliser la même action est présente dans la page.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.1.1 Keyboard (A) / [2.1.1 Clavier](#)
- 9.2.1.2 No Keyboard Trap (A) / [2.1.2 Pas de piège au clavier](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G4](#) – [G90](#)

Critère 4.13 Chaque [média temporel](#) et [non temporel](#) est-il [compatible avec les technologies d'assistance](#) (hors cas particuliers) ?

- Test 4.13.1 : Chaque [média temporel](#) et [non temporel](#) vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - Le nom, le rôle, la valeur, le paramétrage et les changements d'états des composants d'interfaces sont accessibles aux technologies d'assistance via une API d'accessibilité ;
 - Une alternative [compatible avec une API d'accessibilité](#) permet d'accéder aux mêmes fonctionnalités.
- Test 4.13.2 : Chaque [média temporel](#) et [non temporel](#) qui possède une alternative [compatible avec les technologies d'assistance](#), vérifie-t-il une de ces conditions ?
 - L'alternative est adjacente au [média temporel](#) ou [non temporel](#) ;
 - L'alternative est accessible via un [lien ou bouton adjacent](#) ;
 - Un mécanisme permet de remplacer le [média temporel](#) ou [non temporel](#) par son alternative.

Cas particuliers

Il existe une gestion de cas particulier lorsque le [média temporel](#) ou [non temporel](#) est utilisé à des fins décoratives (c'est-à-dire qu'il n'apporte aucune information).

Dans ces situations, le critère est non applicable.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.4.1.2 Name, role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F15](#) – [F54](#) – [G10](#) – [G135](#)

2.2.5 5. Tableaux

Critère 5.1 Chaque [tableau de données complexe](#) a-t-il un [résumé](#) ?

- Test 5.1.1 : Pour chaque [tableau de données complexe](#), un [résumé](#) est-il disponible ?

Note technique

La spécification HTML propose plusieurs [méthodes pour lier un résumé à un tableau](#) (tableau lié à un passage de texte avec l'attribut `aria-describedby`, tableau groupé dans un élément `figure` avec un résumé présent dans un élément `figcaption` ou un élément `p`, résumé présent dans un élément `details` contenu dans l'élément `caption`). Ces méthodes n'ont pas un support suffisant pour être utilisées actuellement.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [H73](#)

Critère 5.2 Pour chaque [tableau de données complexe](#) ayant un [résumé](#), celui-ci est-il pertinent ?

- Test 5.2.1 : Pour chaque [tableau de données complexe](#) ayant un [résumé](#), celui-ci est-il pertinent ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [H73](#)

Critère 5.3 Pour chaque [tableau de mise en forme](#), le contenu linéarisé reste-t-il compréhensible ?

- Test 5.3.1 : Chaque [tableau de mise en forme](#) vérifie-t-il ces conditions ?
 - Le contenu linéarisé reste compréhensible ;
 - La balise `<table>` possède un attribut `role="presentation"`.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.2 Meaningful Sequence (A) / [1.3.2 Ordre séquentiel logique](#)
- 9.4.1.2 Name, Role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA4 – F49](#)

Critère 5.4 Pour chaque [tableau de données ayant un titre](#), le titre est-il correctement associé au tableau de données ?

- Test 5.4.1 : Pour chaque [tableau de données ayant un titre](#), le titre est-il correctement associé au tableau de données ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [H39](#)

Critère 5.5 Pour chaque [tableau de données ayant un titre](#), celui-ci est-il pertinent ?

- Test 5.5.1 : Pour chaque [tableau de données ayant un titre](#), ce titre permet-il d'identifier le contenu du [tableau de données](#) de manière claire et concise ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 91.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [H39](#)

Critère 5.6 Pour chaque [tableau de données](#), chaque [en-tête de colonne](#) et chaque [en-tête de ligne](#) sont-ils correctement déclarés ?

- Test 5.6.1 : Pour chaque [tableau de données](#), chaque [en-tête de colonne](#) s'appliquant à la totalité de la colonne vérifie-t-il une de ces conditions ?
 - L'[en-tête de colonnes](#) est structuré au moyen d'une balise `<th>` ;
 - L'[en-tête de colonnes](#) est structuré au moyen d'une balise pourvue d'un attribut WAI-ARIA `role="columnheader"`.
- Test 5.6.2 : Pour chaque [tableau de données](#), chaque [en-tête de ligne](#) s'appliquant à la totalité de la ligne vérifie-t-il une de ces conditions ?
 - L'[en-tête de lignes](#) est structuré au moyen d'une balise `<th>` ;
 - L'[en-tête de lignes](#) est structuré au moyen d'une balise pourvue d'un attribut WAI-ARIA `role="rowheader"`.
- Test 5.6.3 : Pour chaque [tableau de données](#), chaque en-tête ne s'appliquant pas à la totalité de la ligne ou de la colonne est-il structuré au moyen d'une balise `<th>` ?
- Test 5.6.4 : Pour chaque [tableau de données](#), chaque cellule associée à plusieurs en-têtes est-elle structurée au moyen d'une balise `<td>` ou `<th>` ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 91.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F91](#) – [H51](#)

Critère 5.7 Pour chaque [tableau de données](#), la technique appropriée permettant d'associer chaque cellule avec ses [en-têtes](#) est-elle utilisée (hors cas particuliers) ?

- Test 5.7.1 : Pour chaque contenu de balise `<th>` s'appliquant à la totalité de la ligne ou de la colonne, la balise `<th>` respecte-t-elle une de ces conditions (hors cas particuliers) ?
 - La balise `<th>` possède un attribut `id` unique ;
 - La balise `<th>` possède un attribut `scope` ;
 - La balise `<th>` possède un attribut WAI-ARIA `role="rowheader"` OU `role="columnheader"`.
- Test 5.7.2 : Pour chaque contenu de balise `<th>` s'appliquant à la totalité de la ligne ou de la colonne et possédant un attribut `scope`, la balise `<th>` vérifie-t-elle une de ces conditions ?

- La balise <th> possède un attribut `scope` avec la valeur "row" pour les [en-têtes de ligne](#) ;
- La balise <th> possède un attribut `scope` avec la valeur "col" pour les [en-têtes de colonne](#).
- Test 5.7.3 : Pour chaque contenu de balise <th> ne s'appliquant pas à la totalité de la ligne ou de la colonne, la balise <th> vérifie-t-elle ces conditions ?
 - La balise <th> ne possède pas d'attribut `scope` ;
 - La balise <th> ne possède pas d'attribut WAI-ARIA `role="rowheader"` ou `role="columnheader"` ;
 - La balise <th> possède un attribut `id` unique.
- Test 5.7.4 : Pour chaque contenu de balise <td> ou <th> associée à un ou plusieurs en-têtes possédant un attribut `id`, la balise vérifie-t-elle ces conditions ?
 - La balise possède un attribut `headers` ;
 - L'attribut `headers` possède la liste des valeurs d'attribut `id` des [en-têtes](#) associés.
- Test 5.7.5 : Pour chaque balise pourvue d'un attribut WAI-ARIA `role="rowheader"` ou `role="columnheader"` dont le contenu s'applique à la totalité de la ligne ou de la colonne, la balise vérifie-t-elle une de ces conditions ?
 - La balise possède un attribut WAI-ARIA `role="rowheader"` pour les [en-têtes de ligne](#) ;
 - La balise possède un attribut WAI-ARIA `role="columnheader"` pour les [en-têtes de colonne](#).

Cas particulier

Dans le cas de tableaux de données ayant des en-têtes sur une seule ligne ou une seule colonne, les en-têtes peuvent être structurés à l'aide de balise <th> sans attribut `scope`.

Note technique

Si l'attribut `headers` est implémenté sur une cellule déjà reliée à un en-tête (de ligne ou de colonne) avec l'attribut `scope` (avec la valeur `col` ou `row`), c'est l'en-tête ou les en-têtes référencés par l'attribut `headers` qui seront restitués aux technologies d'assistance. Les en-têtes reliés avec l'attribut `scope` seront ignorés.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F90](#) – [H43](#) – [H63](#)

Critère 5.8 Chaque [tableau de mise en forme](#) ne doit pas utiliser d'éléments propres aux [tableaux de données](#). Cette règle est-elle respectée ?

- Test 5.8.1 : Chaque [tableau de mise en forme](#) (balise <table>) vérifie-t-il ces conditions ?
 - Le tableau de mise en forme (balise <table>) n'a pas d'attribut `summary` (sinon vide) et ne contient pas de balises <caption>, <th>, <thead>, <tfoot> ou de balises ayant un attribut WAI-ARIA `role="rowheader"`, `role="columnheader"` ;
 - Les cellules du tableau de mise en forme (balises <td>) ne possèdent pas d'attributs `scope`, `headers` et `axis`.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F46](#)

2.2.6 6. Liens

Critère 6.1 Chaque [lien](#) est-il explicite (hors cas particuliers) ?

- Test 6.1.1 : Chaque [lien texte](#) vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - L'[intitulé de lien](#) seul permet d'en comprendre la fonction et la destination ;
 - L'[intitulé de lien](#) additionné au [contexte du lien](#) permet d'en comprendre la fonction et la destination.
- Test 6.1.2 : Chaque [lien image](#) vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - L'[intitulé de lien](#) seul permet d'en comprendre la fonction et la destination ;
 - L'[intitulé de lien](#) additionné au [contexte du lien](#) permet d'en comprendre la fonction et la destination.
- Test 6.1.3 : Chaque [lien composite](#) vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - L'[intitulé de lien](#) seul permet d'en comprendre la fonction et la destination ;
 - L'[intitulé de lien](#) additionné au [contexte du lien](#) permet d'en comprendre la fonction et la destination.
- Test 6.1.4 : Chaque [lien SVG](#) vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - L'[intitulé de lien](#) seul permet d'en comprendre la fonction et la destination ;
 - L'[intitulé de lien](#) additionné au [contexte du lien](#) permet d'en comprendre la fonction et la destination.
- Test 6.1.5 : Pour chaque [lien](#) ayant un [intitulé visible](#), le [nom accessible du lien](#) contient-il au moins l'[intitulé visible](#) (hors cas particuliers) ?

Cas particuliers

Il existe une gestion de cas particuliers pour les tests 6.1.1, 6.1.2, 6.1.3 et 6.1.4 lorsque le lien est [ambigu pour tout le monde](#). Dans cette situation, où il n'est pas possible de rendre le lien explicite dans son contexte, le critère est non applicable.

Il existe une gestion de cas particuliers pour le test 6.1.5 lorsque :

- La ponctuation et les lettres majuscules sont présentes dans le texte de l'[intitulé visible](#) : elles peuvent être ignorées dans le nom accessible sans porter à conséquence ;
- Le texte de l'[intitulé visible](#) sert de symbole : le texte ne doit pas être interprété littéralement au niveau du nom accessible. Le nom doit exprimer la fonction véhiculée par le symbole (par exemple, "B" au niveau d'un éditeur de texte aura pour nom accessible "Mettre en gras", le signe ">" en fonction du contexte signifiera "Suivant" ou "Lancer la vidéo"). Le cas des symboles mathématiques fait cependant exception (voir la note ci-dessous).

Note : si l'étiquette visible représente une expression mathématique, les symboles mathématiques peuvent être repris littéralement pour servir d'étiquette au nom accessible (ex. : "A>B"). Il est laissé à l'utilisateur le soin d'opérer la correspondance entre l'expression et ce qu'il doit épeler compte tenu de la connaissance qu'il a du fonctionnement de son logiciel de saisie vocale ("A plus grand que B" ou "A supérieur à B").

Notes techniques

Lorsque l'intitulé visible est complété par une autre expression dans le nom accessible :

- WCAG insiste sur le placement de l'intitulé visible au début du nom accessible sans toutefois réserver l'exclusivité de cet emplacement ;
- WCAG considère comme un cas d'échec une correspondance non exacte de la chaîne de caractères de l'intitulé visible au sein du nom accessible.

Par exemple, si l'on considère l'intitulé visible « Commander maintenant » complété dans le nom accessible par l'expression « produit X », on peut avoir les différents cas suivants :

- « Commander maintenant produit X » est valide (bonne pratique) ;
- « Produit X : commander maintenant » est valide ;
- « Commander produit X maintenant » est non valide.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)
- 9.2.4.4 Link Purpose (In Context) (A) / [2.4.4 Fonction du lien \(selon le contexte\)](#)
- 9.2.5.3 Label in Name (A) / [2.5.3 Étiquette dans le nom](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA7](#) – [ARIA8](#) – [F63](#) – [F89](#) – [G53](#) – [G91](#) – [H30](#) – [H78](#) – [H79](#) – [H80](#) – [H81](#)

Critère 6.2 Dans chaque page web, chaque [lien](#) a-t-il un [intitulé](#) ?

- Test 6.2.1 : Dans chaque page web, chaque [lien](#) a-t-il un [intitulé](#) entre `<a>` et `` ?

Note technique

Une ancre n'est pas un lien même si pendant longtemps l'élément `<a>` a servi de support à cette technique. Elle n'est donc pas concernée par le présent critère.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)
- 9.2.4.4 Link Purpose (In Context) (A) / [2.4.4 Fonction du lien \(selon le contexte\)](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F89](#) – [G91](#) – [H30](#)

2.2.7 7. Scripts

Critère 7.1 Chaque [script](#) est-il, si nécessaire, [compatible avec les technologies d'assistance](#) ?

- Test 7.1.1 : Chaque [script](#) qui génère ou contrôle un [composant d'interface](#) vérifie-t-il, si nécessaire, une de ces conditions ?
 - Le [nom, le rôle, la valeur, le paramétrage et les changements d'états](#) sont accessibles aux technologies d'assistance via une API d'accessibilité ;
 - Un [composant d'interface](#) accessible permettant d'accéder aux mêmes fonctionnalités est présent dans la page ;
 - Une [alternative](#) accessible permet d'accéder aux mêmes fonctionnalités.
- Test 7.1.2 : Chaque [script](#) qui génère ou contrôle un [composant d'interface](#) respecte-t-il une de ces conditions ?
 - Le [composant d'interface](#) est [correctement restitué](#) par les technologies d'assistance ;
 - Une [alternative](#) accessible permet d'accéder aux mêmes fonctionnalités.
- Test 7.1.3 : Chaque [script](#) qui génère ou contrôle un [composant d'interface](#) vérifie-t-il ces conditions (hors cas particuliers) ?
 - Le composant possède un nom pertinent ;
 - Le nom accessible du composant contient au moins l'[intitulé visible](#) ;
 - Le composant possède un rôle pertinent.

Cas particuliers

Il existe une gestion de cas particuliers pour le test 71.3 lorsque :

- La ponctuation et les lettres majuscules sont présentes dans le texte de l'intitulé visible : elles peuvent être ignorées dans le nom accessible sans porter à conséquence ;
- Le texte de l'intitulé visible sert de symbole : le texte ne doit pas être interprété littéralement au niveau du nom accessible. Le nom doit exprimer la fonction véhiculée par le symbole (par exemple, "B" au niveau d'un éditeur de texte aura pour nom accessible "Mettre en gras", le signe ">" en fonction du contexte signifiera "Suivant" ou "Lancer la vidéo"). Le cas des symboles mathématiques fait cependant exception (voir la note ci-dessous).

Note : si l'étiquette visible représente une expression mathématique, les symboles mathématiques peuvent être repris littéralement pour servir d'étiquette au nom accessible (ex. : "A>B"). Il est laissé à l'utilisateur le soin d'opérer la correspondance entre l'expression et ce qu'il doit épeler compte tenu de la connaissance qu'il a du fonctionnement de son logiciel de saisie vocale ("A plus grand que B" ou "A supérieur à B").

Notes techniques

Le critère 7.1 implémente la notion de « compatible avec les technologies d'assistance » telle que définie par les WCAG, ainsi que le recours à WAI-ARIA pour rendre un composant ou une fonctionnalité accessible. Le bon usage de WAI-ARIA est vérifié via les tests 7.1.1, 7.1.2, 7.1.3.

Note importante : dans un environnement HTML5, beaucoup de composants peuvent nécessiter JavaScript pour fonctionner ; en conséquence la fourniture d'une alternative à un composant JavaScript qui ne pourrait pas être rendu accessible devra bénéficier d'une méthode spécifique au composant en cause, permettant de le remplacer par une alternative accessible (et de le réactiver). Cela signifie que la désactivation de JavaScript pour l'ensemble de la page ne sera pas acceptée comme une méthode valable, à moins qu'elle ne remette pas en cause l'utilisation des autres composants.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.5.3 Label in Name (A) / [2.5.3 Étiquette dans le nom](#)
- 9.4.1.2 Name, Role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA4](#) – [ARIA5](#) – [ARIA18](#) – [ARIA19](#) – [F15](#) – [F19](#) – [F20](#) – [F42](#) – [F59](#) – [F79](#) – [G10](#) – [G135](#) – [G136](#) – [SCR21](#)

Critère 7.2 Pour chaque [script](#) ayant une [alternative](#), cette alternative est-elle pertinente ?

- Test 7.2.1 : Chaque [script](#) débutant par la balise `<script>` et ayant une [alternative](#) vérifie-t-il une de ces conditions ?
 - L'[alternative](#) entre `<noscript>` et `</noscript>` permet d'accéder à des contenus et des fonctionnalités similaires ;

- La page affichée, lorsque JavaScript est désactivé, permet d'accéder à des contenus et des fonctionnalités similaires ;
 - La page alternative permet d'accéder à des contenus et des fonctionnalités similaires ;
 - Le langage de script côté serveur permet d'accéder à des contenus et des fonctionnalités similaires ;
 - L'alternative présente dans la page permet d'accéder à des contenus et des fonctionnalités similaires.
- Test 7.2.2 : Chaque élément non textuel mis à jour par un [script](#) (dans la page, ou dans un [cadre](#)) et ayant une [alternative](#) vérifie-t-il ces conditions ?
 - L'alternative de l'élément non textuel est mise à jour ;
 - L'alternative mise à jour est pertinente.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)
- 9.4.1.2 Name, Role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F19](#) – [F20](#) – [G136](#)

Critère 7.3 Chaque [script](#) est-il [contrôlable par le clavier et par tout dispositif de pointage](#) (hors cas particuliers) ?

- Test 7.3.1 : Chaque élément possédant un gestionnaire d'événement contrôlé par un script vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - L'élément est [accessible par le clavier et tout dispositif de pointage](#) ;
 - Un élément [accessible par le clavier et tout dispositif de pointage](#) permettant de réaliser la même action est présent dans la page.
- Test 7.3.2 : Un [script](#) ne doit pas supprimer le focus d'un élément qui le reçoit. Cette règle est-elle respectée (hors cas particuliers) ?

Cas particulier

Il existe une gestion de cas particuliers lorsque la fonctionnalité dépend de l'utilisation d'un gestionnaire d'événement sans équivalent universel ; par exemple, une application de dessin à main levée ne pourra pas être rendue contrôlable au clavier. Dans ces situations, le critère est non applicable.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)
- 9.2.1.1 Keyboard (A) / [2.1.1 Clavier](#)
- 9.2.4.7 Focus Visible (AA) / [2.4.7 Visibilité du focus](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F42](#) – [F54](#) – [F55](#) – [G90](#) – [G202](#) – [SCR2](#) – [SCR20](#) – [SCR29](#) – [SCR35](#)

Critère 7.4 Pour chaque [script](#) qui initie un [changement de contexte](#), l'utilisateur est-il averti ou en a-t-il le contrôle ?

- Test 7.4.1 : Chaque [script](#) qui initie un [changement de contexte](#) vérifie-t-il une de ces conditions ?
 - L'utilisateur est averti par un texte de l'action du script et du type de changement avant son déclenchement ;
 - Le changement de contexte est initié par un bouton (input de type `submit`, `button` ou image ou balise `<button>`) explicite ;
 - Le changement de contexte est initié par un lien explicite.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.3.2.1 On Focus (A) / [3.2.1 Au focus](#)
- 9.3.2.2 On Input (A) / [3.2.2 À la saisie](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F9](#) – [F22](#) – [F36](#) – [F37](#) – [F41](#) – [G13](#) – [G76](#) – [G80](#) – [G107](#) – [H32](#) – [H84](#) – [SCR19](#)

Critère 7.5 Dans chaque page web, les [messages de statut](#) sont-ils correctement restitués par les technologies d'assistance ?

- Test 7.5.1 : Chaque [message de statut](#) qui informe de la réussite, du résultat d'une action ou bien de l'état d'une application utilise-t-il l'attribut WAI-ARIA `role="status"` ?
- Test 7.5.2 : Chaque [message de statut](#) qui présente une suggestion, ou avertit de l'existence d'une erreur utilise-t-il l'attribut WAI-ARIA `role="alert"` ?
- Test 7.5.3 : Chaque [message de statut](#) qui indique la progression d'un processus utilise-t-il l'un des attributs WAI-ARIA `role="log"`, `role="progressbar"` ou `role="status"` ?

Notes techniques

Les rôles WAI-ARIA `log`, `status` et `alert` ont implicitement une valeur d'attribut WAI-ARIA `aria-live` et `aria-atomic`. On pourra donc considérer (conformément à la spécification WAI-ARIA 1.1) que :

- Un attribut WAI-ARIA `aria-live="polite"` associé à un message de statut peut valoir pour un rôle WAI-ARIA `log` ;

- Un attribut WAI-ARIA `aria-live="polite"` et un attribut WAI-ARIA `aria-atomic="true"` associés à un message de statut peuvent valoir pour un rôle WAI-ARIA `status` ;
- Un attribut WAI-ARIA `aria-live="assertive"` et un attribut WAI-ARIA `aria-atomic="true"` associés à un message de statut peuvent valoir pour un rôle WAI-ARIA `alert`.

C'est sous réserve que la nature du message de statut satisfasse bien à la correspondance implicitement établie. Dans le cas d'un message de statut indiquant la progression d'un processus et matérialisé graphiquement par une barre de progression, un rôle WAI-ARIA `progressbar` explicite est nécessaire.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.4.1.3 Status Messages (AA) / [4.1.3 Messages d'état](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA19](#) – [ARIA22](#) – [ARIA23](#)

2.2.8 8. Éléments obligatoires

Critère 8.1 Chaque page web est-elle définie par un type de document ?

- Test 8.1.1 : Pour chaque page web, le type de document (balise `doctype`) est-il présent ?
- Test 8.1.2 : Pour chaque page web, le type de document (balise `doctype`) est-il valide ?
- Test 8.1.3 : Pour chaque page web possédant une déclaration de type de document, celle-ci est-elle située avant la balise `<html>` dans le code source ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.4.1.1 Parsing (A) / [4.1.1 Analyse syntaxique](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G134](#) – [G192](#)

Critère 8.2 Pour chaque page web, le code source généré est-il valide selon le type de document spécifié ?

- Test 8.2.1 : Pour chaque déclaration de type de document, le code source généré de la page vérifie-t-il ces conditions ?
 - Les balises, attributs et valeurs d'attributs respectent les [règles d'écriture](#) ;
 - L'imbrication des balises est conforme ;
 - L'ouverture et la fermeture des balises sont conformes ;
 - Les valeurs d'attribut `id` sont uniques dans la page ;
 - Les attributs ne sont pas doublés sur un même élément.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.4.1.1 Parsing (A) / [4.1.1 Analyse syntaxique](#)
- 9.4.1.2 Name, Role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F70](#) – [F77](#) – [H74](#) – [H93](#) – [H94](#)

Critère 8.3 Dans chaque page web, la [langue par défaut](#) est-elle présente ?

- Test 8.3.1 : Pour chaque page web, l'indication de langue par défaut vérifie-t-elle une de ces conditions ?
 - L'indication de la langue de la page (attribut `lang` et/ou `xml:lang`) est donnée pour l'élément `html` ;
 - L'indication de la langue de la page (attribut `lang` et/ou `xml:lang`) est donnée sur chaque élément de texte ou sur l'un des éléments parents.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.3.1.1 Language of Page (A) / [3.1.1 Langue de la page](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [H57](#)

Critère 8.4 Pour chaque page web ayant une [langue par défaut](#), le [code de langue](#) est-il pertinent ?

- Test 8.4.1 : Pour chaque page web ayant une langue par défaut, le code de langue vérifie-t-il ces conditions ?
 - Le code de langue est valide ;
 - Le code de langue est pertinent.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.3.1.1 Language of Page (A) / [3.1.1 Langue de la page](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [H57](#)

Critère 8.5 Chaque page web a-t-elle un [titre de page](#) ?

- Test 8.5.1 : Chaque page web a-t-elle un [titre de page](#) (balise `<title>`) ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.4.2 Page Titled (A) / [2.4.2 Titre de page](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G88](#) – [G127](#) – [H25](#)

Critère 8.6 Pour chaque page web ayant un [titre de page](#), ce titre est-il pertinent ?

- Test 8.6.1 : Pour chaque page web ayant un [titre de page](#) (balise <title>), le contenu de cette balise est-il pertinent ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.4.2 Page Titled (A) / [2.4.2 Titre de page](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G88](#) – [G127](#) – [H25](#)

Critère 8.7 Dans chaque page web, chaque [changement de langue](#) est-il indiqué dans le code source (hors cas particuliers) ?

- Test 8.7.1 : Dans chaque page web, chaque texte écrit dans une langue différente de la [langue par défaut](#) vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - L'indication de langue est donnée sur l'élément contenant le texte (attribut `lang` et/ou `xml:lang`) ;
 - L'indication de langue est donnée sur un des éléments parents (attribut `lang` et/ou `xml:lang`)

Cas particuliers

Il y a une gestion de cas particuliers sur le changement de langue pour les cas suivants :

- Nom propre, le critère est non applicable ;
- Nom commun de langue étrangère présent dans le dictionnaire officiel de la langue (voir note 1 ci-dessous) par défaut de la page web, le critère est non applicable ;
- Le terme de langue étrangère soumis, via un [champ de formulaire](#) et rappelé dans la page (par exemple comme indication du terme recherché dans le cas d'un moteur de recherche), le critère est non applicable ;
- Passage de texte dont la langue ne peut pas être déterminée : le critère est non applicable ;
- Terme ou passage de texte issus d'une langue morte ou imaginaire pour laquelle il n'existe pas d'interprétation vocale : le critère est non applicable.

Note 1 : le dictionnaire officiel est celui recommandé par l'académie en charge de la langue en question. Pour la France, par exemple, le lien vers le dictionnaire officiel se trouve sur le site de l'Académie française à l'adresse suivante : <http://www.academie-francaise.fr/le-dictionnaire/la-9e-edition>. Pour toute demande auprès du service du dictionnaire de l'Académie française, utiliser le formulaire de contact du service du dictionnaire.

Note 2 : pour les noms communs de langue étrangère, absents dans le dictionnaire officiel de la langue par défaut de la page web, et qui sont passés dans le langage commun (exemple : newsletter) : le critère est applicable, uniquement lorsque l'absence d'indication de langue peut provoquer une incompréhension pour la restitution.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.3.1.2 Language of Parts (AA) / [3.1.2 Langue d'un passage](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [H58](#)

Critère 8.8 Dans chaque page web, le code de langue de chaque [changement de langue](#) est-il valide et pertinent ?

- Test 8.8.1 : Pour chaque page web, le code de langue de chaque [changement de langue](#) vérifie-t-il ces conditions ?
 - Le code de langue est valide ;
 - Le code de langue est pertinent.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.3.1.2 Language of Parts (AA) / [3.1.2 Langue d'un passage](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [H58](#)

Critère 8.9 Dans chaque page web, les balises ne doivent pas être utilisées [uniquement à des fins de présentation](#). Cette règle est-elle respectée ?

- Test 8.9.1 : Dans chaque page web les balises (à l'exception de `<div>`, `` et `<table>`) ne doivent pas être utilisées [uniquement à des fins de présentation](#). Cette règle est-elle respectée ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F43](#) – [F92](#) – [G115](#) – [H88](#)

Critère 8.10 Dans chaque page web, les changements du [sens de lecture](#) sont-ils signalés ?

- Test 8.10.1 : Dans chaque page web, chaque texte dont le sens de lecture est différent du [sens de lecture](#) par défaut est contenu dans une balise possédant un attribut `dir` ?
- Test 8.10.2 : Dans chaque page web, chaque changement du [sens de lecture](#) (attribut `dir`) vérifie-t-il ces conditions ?

- La valeur de l'attribut `dir` est conforme (`rtl` ou `ltr`) ;
- La valeur de l'attribut `dir` est pertinente.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.2 Meaningful Sequence (A) / [1.3.2 Ordre séquentiel logique](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [H56](#)

2.2.9 9. Structuration de l'information

Critère 9.1 Dans chaque page web, l'information est-elle structurée par l'utilisation appropriée de [titres](#) ?

- Test 9.1.1 : Dans chaque page web, la hiérarchie entre les [titres](#) (balise `<hx>` ou balise possédant un attribut WAI-ARIA `role="heading"` associé à un attribut WAI-ARIA `aria-level`) est-elle pertinente ?
- Test 9.1.2 : Dans chaque page web, le contenu de chaque [titre](#) (balise `<hx>` ou balise possédant un attribut WAI-ARIA `role="heading"` associé à un attribut WAI-ARIA `aria-level`) est-il pertinent ?
- Test 9.1.3 : Dans chaque page web, chaque passage de texte constituant un [titre](#) est-il structuré à l'aide d'une balise `<hx>` ou d'une balise possédant un attribut WAI-ARIA `role="heading"` associé à un attribut WAI-ARIA `aria-level` ?

Note technique

WAI-ARIA permet de définir des titres via le rôle `heading` et l'attribut `aria-level` (indication du niveau de titre). Bien qu'il soit préférable d'utiliser l'élément de titre natif en HTML `<hx>`, l'utilisation du rôle WAI-ARIA `heading` est compatible avec l'accessibilité.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)
- 9.2.4.1 Bypass Blocks (A) / [2.4.1 Contourner des blocs](#)
- 9.2.4.6 Headings and Labels (AA) / [2.4.6 En-têtes et étiquettes](#)
- 9.4.1.2 Name, Role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA4](#) – [ARIA12](#) – [G115](#) – [G130](#) – [G141](#) – [H42](#)

Critère 9.2 Dans chaque page web, la [structure du document](#) est-elle cohérente (hors cas particuliers) ?

- Test 9.2.1 : Dans chaque page web, la [structure du document](#) vérifie-t-elle ces conditions (hors cas particuliers) ?

- La [zone d'en-tête de la page](#) est structurée via une balise `<header>` ;
- Les [zones de navigation principales et secondaires](#) sont structurées via une balise `<nav>` ;
- La balise `<nav>` est réservée à la structuration des [zones de navigation principales et secondaires](#) ;
- La [zone de contenu principal](#) est structurée via une balise `<main>` ;
- La [structure du document](#) utilise une balise `<main>` visible unique ;
- La [zone de pied de page](#) est structurée via une balise `<footer>`.

Cas particulier

Lorsque le doctype déclaré dans la page n'est pas le doctype HTML5, ce critère est non applicable.

Note technique

La balise `<main>` peut être utilisée plusieurs fois dans le même document HTML. Néanmoins, il ne peut y avoir en permanence qu'une seule balise visible et lisible par les technologies d'assistances, les autres devant disposer d'un attribut `hidden` ou d'un style permettant de les masquer aux technologies d'assistances. À noter cependant que l'utilisation d'un style seul restera insuffisante pour assurer l'unicité d'une balise `<main>` visible en cas de désactivation des feuilles de styles.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA11](#) – [G115](#)

Critère 9.3 Dans chaque page web, chaque [liste](#) est-elle correctement structurée ?

- Test 9.3.1 : Dans chaque page web, les informations regroupées visuellement sous forme de [liste](#) non ordonnée vérifient-elles une de ces conditions ?
 - La liste utilise les balises HTML `` et `` ;
 - La liste utilise les attributs WAI-ARIA `role="list"` et `role="listitem"`.
- Test 9.3.2 : Dans chaque page web, les informations regroupées visuellement sous forme de [liste](#) ordonnée vérifient-elles une de ces conditions ?
 - La liste utilise les balises HTML `` et `` ;
 - La liste utilise les attributs WAI-ARIA `role="list"` et `role="listitem"`.
- Test 9.3.3 : Dans chaque page web, les informations regroupées sous forme de [liste](#) de description utilisent-elles les balises `<dl>` et `<dt>/<dd>` ?

Notes techniques

Les attributs WAI-ARIA `role="list"` et `role="listitem"` peuvent nécessiter l'utilisation des attributs WAI-ARIA `aria-setsize` et `aria-posinset` dans le cas où l'ensemble de la liste n'est pas disponible via le DOM généré au moment de la consultation.

Les attributs WAI-ARIA `role="tree"`, `role="tablist"`, `role="menu"`, `role="combobox"` et `role="listbox"` ne sont pas équivalents à une liste HTML `` ou ``.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F2](#) – [G115](#) – [G153](#) – [H40](#) – [H48](#)

Critère 9.4 Dans chaque page web, chaque citation est-elle correctement indiquée ?

- Test 9.4.1 : Dans chaque page web, chaque citation courte utilise-t-elle une balise `<q>` ?
- Test 9.4.2 : Dans chaque page web, chaque bloc de citation utilise-t-il une balise `<blockquote>` ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F2](#) – [G115](#) – [H49](#)

2.2.1010. Présentation de l'information

Critère 10.1 Dans le site web, des [feuilles de styles](#) sont-elles utilisées pour contrôler la [présentation de l'information](#) ?

- Test 10.1.1 : Dans chaque page web, les balises servant à la [présentation de l'information](#) ne doivent pas être présentes dans le code source généré des pages. Cette règle est-elle respectée ?
- Test 10.1.2 : Dans chaque page web, les attributs servant à la [présentation de l'information](#) ne doivent pas être présents dans le code source généré des pages. Cette règle est-elle respectée ?
- Test 10.1.3 : Dans chaque page web, l'utilisation des espaces vérifie-t-elle ces conditions ?
 - Les espaces ne sont pas utilisées pour séparer les lettres d'un mot ;
 - Les espaces ne sont pas utilisées pour simuler des tableaux ;
 - Les espaces ne sont pas utilisées pour simuler des colonnes de texte.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)
- 9.1.3.2 Meaningful Sequence (A) / [1.3.2 Ordre séquentiel logique](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [C6](#) – [C8](#) – [C18](#) – [C22](#) – [F32](#) – [F33](#) – [F34](#) – [F48](#) – [G140](#)

Critère 10.2 Dans chaque page web, le [contenu visible](#) porteur d'information reste-t-il présent lorsque les [feuilles de styles](#) sont désactivées ?

- Test 10.2.1 : Dans chaque page web, l'information reste-t-elle présente lorsque les [feuilles de styles](#) sont désactivées ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)
- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F3](#) – [F87](#) – [G140](#)

Critère 10.3 Dans chaque page web, l'information reste-t-elle [compréhensible](#) lorsque les [feuilles de styles](#) sont désactivées ?

- Test 10.3.1 : Dans chaque page web, l'information reste-t-elle [compréhensible](#) lorsque les [feuilles de styles](#) sont désactivées ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.2 Meaningful Sequence (A) / [1.3.2 Ordre séquentiel logique](#)
- 9.2.4.3 Focus Order (A) / [2.4.3 Parcours du focus](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F1](#) – [G59](#) – [G140](#)

Critère 10.4 Dans chaque page web, le texte reste-t-il lisible lorsque la [taille des caractères](#) est augmentée jusqu'à 200 %, au moins (hors cas particuliers) ?

- Test 10.4.1 : Dans chaque page web, l'augmentation de la [taille des caractères](#) jusqu'à 200 %, au moins, ne doit pas provoquer de perte d'information. Cette règle est-elle respectée selon une de ces conditions (hors cas particuliers) ?
 - Lors de l'utilisation de la fonction d'agrandissement du texte du navigateur ;
 - Lors de l'utilisation des fonctions de zoom graphique du navigateur ;
 - Lors de l'utilisation d'un [composant d'interface](#) propre au site permettant d'agrandir le texte ou de zoomer.

- Test 10.4.2 : Dans chaque page web, l'augmentation de la taille des caractères jusqu'à 200%, au moins, doit être possible pour l'ensemble du texte dans la page. Cette règle est-elle respectée selon une de ces conditions (hors cas particuliers) ?
 - Lors de l'utilisation de la fonction d'agrandissement du texte du navigateur ;
 - Lors de l'utilisation des fonctions de zoom graphique du navigateur ;
 - Lors de l'utilisation d'un [composant d'interface](#) propre au site permettant d'agrandir le texte ou de zoomer.

Cas particuliers

Font exception à ce critère, les contenus pour lesquels l'utilisateur n'a pas de possibilité de personnalisation :

- Les sous-titres incrustés dans une vidéo ;
- Les textes en image ;
- Le texte au sein d'une balise <canvas>.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.4.4 Resize Text (AA) / [1.4.4 Redimensionnement du texte](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [C12](#) – [C13](#) – [C14](#) – [C17](#) – [C28](#) – [F69](#) – [F80](#) – [G146](#) – [G179](#) – [SCR34](#)

Critère 10.5 Dans chaque page web, les déclarations CSS de couleurs de fond d'élément et de police sont-elles correctement utilisées ?

- Test 10.5.1 : Dans chaque page web, chaque déclaration CSS de couleurs de police (`color`), d'un élément susceptible de contenir du texte, est-elle accompagnée d'une déclaration de couleur de fond (`background`, `background-color`), au moins, héritée d'un parent ?
- Test 10.5.2 : Dans chaque page web, chaque déclaration de couleur de fond (`background`, `background-color`), d'un élément susceptible de contenir du texte, est-elle accompagnée d'une déclaration de couleur de police (`color`) au moins, héritée d'un parent ?
- Test 10.5.3 : Dans chaque page web, chaque utilisation d'une image pour créer une couleur de fond d'un élément susceptible de contenir du texte, via CSS (`background`, `background-image`), est-elle accompagnée d'une déclaration de couleur de fond (`background`, `background-color`), au moins, héritée d'un parent ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.4.3 Contrast (Minimum) (AA) / [1.4.3 Contraste \(minimum\)](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F24](#)

Critère 10.6 Dans chaque page web, chaque [lien dont la nature n'est pas évidente](#) est-il visible par rapport au texte environnant ?

- Test 10.6.1 : Dans chaque page web, chaque [lien texte](#) signalé uniquement par la couleur, et dont la nature n'est pas évidente, vérifie-t-il ces conditions ?
 - La couleur du lien a un rapport de [contraste](#) supérieur ou égal à 3:1 par rapport au texte environnant ;
 - Le lien dispose d'une indication visuelle au survol autre qu'un changement de couleur ;
 - Le lien dispose d'une indication visuelle au focus autre qu'un changement de couleur.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.4.1 Use of Color (A) / [1.4.1 Utilisation de la couleur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F73](#) – [G183](#)

Critère 10.7 Dans chaque page web, pour chaque élément recevant le focus, la [prise de focus](#) est-elle visible ?

- Test 10.7.1 : Pour chaque élément recevant le focus, la [prise de focus](#) vérifie-t-elle une de ces conditions ?
 - Le style du focus natif du navigateur n'est pas supprimé ou dégradé ;
 - Un style du focus défini par l'auteur est visible.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.4.1 Use of Color (A) / [1.4.1 Utilisation de la couleur](#)
- 9.2.4.7 Focus Visible (AA) / [2.4.7 Visibilité du focus](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [C15](#) – [F73](#) – [F78](#) – [G149](#) – [G165](#) – [G183](#) – [G195](#) – [SCR31](#)

Critère 10.8 Pour chaque page web, les [contenus cachés](#) ont-ils vocation à être ignorés par les technologies d'assistance ?

- Test 10.8.1 : Dans chaque page web, chaque contenu caché vérifie-t-il une de ces conditions ?
 - Le [contenu caché](#) a vocation à être ignoré par les technologies d'assistance ;

- Le [contenu caché](#) n'a pas vocation à être ignoré par les technologies d'assistance et est rendu restituable par les technologies d'assistance suite à une action de l'utilisateur réalisable au clavier ou par tout dispositif de pointage sur un élément précédent le contenu caché ou suite à un repositionnement du focus dessus.

Notes techniques

WAI-ARIA propose un attribut `aria-hidden` (`true` ou `false`) qui permet d'inhiber la restitution d'un contenu en direction des technologies d'assistance, sans action sur sa visibilité en direction des agents utilisateurs : un contenu avec `aria-hidden="true"` ne sera donc plus vocalisable, mais restera visible.

Sauf si le contenu contrôlé par `aria-hidden` n'a pas vocation à être restitué par les technologies d'assistance, la valeur de l'attribut `aria-hidden` doit être cohérente avec l'état affiché ou masqué du contenu à l'écran.

La spécification HTML5 propose un attribut `hidden` qui permet de rendre indisponible (quand l'attribut `hidden` est présent) un contenu dans le DOM généré (de manière similaire au `type="hidden"` sur un contrôle de formulaire).

Il est possible d'avoir des situations où un contenu contrôlé par `hidden` ou `aria-hidden` se trouve momentanément dans un état incohérent avec le statut affiché ou masqué du contenu, par exemple si l'on désire rendre disponible un élément, mais que son affichage à l'écran reste dépendant d'une action ultérieure. Dans ce cas, c'est l'état final du contenu qui doit être considéré.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.2 Meaningful Sequence (A) / [1.3.2 Ordre séquentiel logique](#)
- 9.4.1.2 Name, Role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G57](#)

Critère 10.9 Dans chaque page web, l'information ne doit pas être donnée uniquement [par la forme, taille ou position](#). Cette règle est-elle respectée ?

- Test 10.9.1 : Dans chaque page web, pour chaque texte ou ensemble de textes, l'information ne doit pas être donnée uniquement [par la forme, taille ou position](#). Cette règle est-elle respectée ?
- Test 10.9.2 : Dans chaque page web, pour chaque image ou ensemble d'images, l'information ne doit pas être donnée uniquement [par la forme, taille ou position](#). Cette règle est-elle respectée ?
- Test 10.9.3 : Dans chaque page web, pour chaque [média temporel](#), l'information ne doit pas être donnée uniquement [par la forme, taille ou position](#). Cette règle est-elle respectée ?

- Test 10.9.4 : Dans chaque page web, pour chaque [média non temporel](#), l'information ne doit pas être donnée uniquement [par la forme, taille ou position](#). Cette règle est-elle respectée ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.3 Sensory Characteristics (A) / [1.3.3 Caractéristiques sensorielles](#)
- 9.1.4.1 Use of Color (A) / [1.4.1 Utilisation de la couleur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F14](#) – [F26](#) – [G96](#) – [G140](#)

Critère 10.10 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 ?

- Test 10.10.1 : Dans chaque page web, pour chaque texte ou ensemble de textes, l'information ne doit pas être donnée uniquement [par la forme, taille ou position](#). Cette règle est-elle implémentée de façon pertinente ?
- Test 10.10.2 : Dans chaque page web, pour chaque image ou ensemble d'images, l'information ne doit pas être donnée uniquement [par la forme, taille ou position](#). Cette règle est-elle implémentée de façon pertinente ?
- Test 10.10.3 : Dans chaque page web, pour chaque [média temporel](#), l'information ne doit pas être donnée uniquement [par la forme, taille ou position](#). Cette règle est-elle implémentée de façon pertinente ?
- Test 10.10.4 : Dans chaque page web, pour chaque [média non temporel](#), l'information ne doit pas être donnée uniquement [par la forme, taille ou position](#). Cette règle est-elle implémentée de façon pertinente ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.3 Sensory Characteristics (A) / [1.3.3 Caractéristiques sensorielles](#)
- 9.1.4.1 Use of Color (A) / [1.4.1 Utilisation de la couleur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F14](#) – [F26](#) – [G96](#) – [G140](#)

Critère 10.11 Pour chaque page web, les contenus peuvent-ils être présentés sans perte d'information ou de fonctionnalité et sans avoir recours soit à un défilement vertical pour une fenêtre ayant une hauteur de 256 px, soit à un défilement horizontal pour une fenêtre ayant une largeur de 320 px (hors cas particuliers) ?

- Test 10.11.1 : Pour chaque page web, lorsque le contenu dont le sens de lecture est horizontal est affiché dans une fenêtre réduite à une largeur de 320 px, l'ensemble des informations et des fonctionnalités sont-elles disponibles sans aucun défilement horizontal (hors cas particuliers) ?

- Test 10.11.2 : Pour chaque page web, lorsque le contenu dont le sens de lecture est vertical est affiché dans une fenêtre réduite à une hauteur de 256 px, l'ensemble des informations et des fonctionnalités sont-elles disponibles sans aucun défilement vertical (hors cas particuliers) ?

Cas particuliers

L'objectif de ce critère est de garantir un défilement dans une unique direction pour une lecture facilitée selon le sens de l'écriture.

Font exception à ce critère, les contenus dont l'agencement requiert deux dimensions pour être compris ou utilisés comme :

- Les images, les graphiques ou les vidéos ;
- Les jeux (jeux de plateforme, par exemple) ;
- Les présentations (type diaporama, par exemple) ;
- Les tableaux de données ;
- Les interfaces où il est nécessaire d'avoir un ascenseur horizontal lors de la manipulation de l'interface.

Note : la majorité des navigateurs sur les systèmes d'exploitation sur mobile (Android, iOS) ne gère pas correctement la redistribution en cas de zoom. Dans ce contexte, le critère sera considéré comme non applicable sur ces environnements.

Note technique

Lorsqu'il est ici question de pixel, il s'agit du pixel CSS tel que défini par le W3C <https://www.w3.org/TR/css3-values/>

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.4.10 Reflow (AA) / [1.4.10 Redistribution](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [C34](#) – [C37](#)

Critère 10.12 Dans chaque page web, les propriétés d'espacement du texte peuvent-elles être redéfinies par l'utilisateur sans perte de contenu ou de fonctionnalité (hors cas particuliers) ?

- Test 10.12.1 : Dans chaque page web, le texte reste-t-il lisible lorsque l'affichage est modifié selon ces conditions (hors cas particuliers) ?
 - L'espacement entre les lignes (`line-height`) est augmenté jusqu'à 1,5 fois la taille de la police ;
 - L'espacement suivant les paragraphes (balise `<p>`) est augmenté jusqu'à 2 fois la taille de la police ;

- L'espacement des lettres (*letter-spacing*) est augmenté jusqu'à 0,12 fois la taille de la police ;
- L'espacement des mots (*word-spacing*) est augmenté jusqu'à 0,16 fois la taille de la police.

Cas particuliers

Font exception à ce critère, les contenus pour lesquels l'utilisateur n'a pas de possibilité de personnalisation :

- Les sous-titres directement intégrés à une vidéo ;
- Les images texte ;
- Le texte au sein d'une balise `<canvas>`.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.4.12 Text Spacing (AA) / [1.4.12 Espacement du texte](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [C8](#) – [C21](#) – [C35](#) – [C36](#)

Critère 10.13 Dans chaque page web, les contenus additionnels apparaissant à la prise de focus ou au survol d'un [composant d'interface](#) sont-ils contrôlables par l'utilisateur (hors cas particuliers) ?

- Test 10.13.1 : Chaque contenu additionnel devenant visible à la prise de focus ou au survol d'un [composant d'interface](#) peut-il être masqué par une action de l'utilisateur sans déplacer le focus ou le pointeur de la souris (hors cas particuliers) ?
- Test 10.13.2 : Chaque contenu additionnel qui apparaît au survol d'un [composant d'interface](#) peut-il être survolé par le pointeur de la souris sans disparaître (hors cas particuliers) ?
- Test 10.13.3 : Chaque contenu additionnel qui apparaît à la prise de focus ou au survol d'un [composant d'interface](#) vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - Le contenu additionnel reste visible jusqu'à ce que l'utilisateur retire le pointeur souris ou le focus du contenu additionnel et du [composant d'interface](#) ayant déclenché son apparition ;
 - Le contenu additionnel reste visible jusqu'à ce que l'utilisateur déclenche une action masquant ce contenu sans déplacer le focus ou le pointeur de la souris du [composant d'interface](#) ayant déclenché son apparition ;
 - Le contenu additionnel reste visible jusqu'à ce qu'il ne soit plus valide.

Cas particuliers

Lorsque le contenu additionnel est contrôlé par l'agent utilisateur (par exemple, attribut `title` ou validation native de formulaire) ou correspond à une fenêtre modale conforme au [motif de conception](#) WAI-ARIA `dialog`, le critère 10.13 est non applicable.

Lorsque le contenu additionnel ne masque ou ne remplace aucun contenu porteur d'information, le test 10.13.1 est non applicable.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.4.13 Content on Hover or Focus (AA) / [1.4.13 Contenu au survol ou au focus](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F95](#)

Critère 10.14 Dans chaque page web, les contenus additionnels apparaissant via les styles CSS uniquement peuvent-ils être rendus visibles au clavier et par tout dispositif de pointage ?

- Test 10.14.1 : Dans chaque page web, les contenus additionnels apparaissant au survol d'un [composant d'interface](#) via les styles CSS respectent-ils si nécessaire une de ces conditions ?
 - Les contenus additionnels apparaissent également à l'activation du composant via le clavier et tout dispositif de pointage ;
 - Les contenus additionnels apparaissent également à la prise de focus du composant ;
 - Les contenus additionnels apparaissent également par le biais de l'activation ou de la prise de focus d'un autre composant.
- Test 10.14.2 : Dans chaque page web, les contenus additionnels apparaissant au focus d'un [composant d'interface](#) via les styles CSS respectent-ils si nécessaire une de ces conditions ?
 - Les contenus additionnels apparaissent également à l'activation du composant via le clavier et tout dispositif de pointage ;
 - Les contenus additionnels apparaissent également au survol du composant ;
 - Les contenus additionnels apparaissent également par le biais de l'activation ou du survol d'un autre composant.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.1.1 Keyboard (A) / [2.1.1 Clavier](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G202](#)

2.2.11 11. Formulaires

| Critère 11.1 Chaque [champ de formulaire](#) a-t-il une [étiquette](#) ?

- Test 11.1.1 : Chaque [champ de formulaire](#) vérifie-t-il une de ces conditions ?
 - Le [champ de formulaire](#) possède un attribut WAI-ARIA `aria-labelledby` référant un [passage de texte](#) identifié ;
 - Le [champ de formulaire](#) possède un attribut WAI-ARIA `aria-label` ;
 - Une balise `<label>` ayant un attribut `for` est associée au [champ de formulaire](#) ;
 - Le [champ de formulaire](#) possède un attribut `title` ;
 - Un bouton adjacent au [champ de formulaire](#) lui fournit une étiquette visible et un élément `<label>` visuellement caché ou un attribut WAI-ARIA `aria-label`, `aria-labelledby` ou `title` lui fournit un nom accessible.
- Test 11.1.2 : Chaque [champ de formulaire](#) associé à une balise `<label>` ayant un attribut `for`, vérifie-t-il ces conditions ?
 - Le [champ de formulaire](#) possède un attribut `id` ;
 - La valeur de l'attribut `for` est égale à la valeur de l'attribut `id` du [champ de formulaire](#) associé.
- Test 11.1.3 : Chaque [champ de formulaire](#) ayant une [étiquette](#) dont le contenu n'est pas visible ou à proximité (masqué, `aria-label`) ou qui n'est pas [accolé](#) au champ (`aria-labelledby`), vérifie-t-il une de ses conditions ?
 - Le [champ de formulaire](#) possède un attribut `title` dont le contenu permet de comprendre la nature de la saisie attendue ;
 - Le [champ de formulaire](#) est accompagné d'un [passage de texte](#) accolé au champ qui devient visible à la prise de focus permettant de comprendre la nature de la saisie attendue ;
 - Le [champ de formulaire](#) est accompagné d'un [passage de texte](#) visible accolé au champ permettant de comprendre la nature de la saisie attendue.

| Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)
- 9.2.4.6 Headings and Labels (AA) / [2.4.6 En-têtes et étiquettes](#)
- 9.3.3.2 Labels or Instructions (A) / [3.3.2 Étiquettes ou instructions](#)
- 9.4.1.2 Name, Role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA6](#) – [ARIA9](#) – [ARIA14](#) – [ARIA16](#) – [F68](#) – [F82](#) – [F86](#) – [G82](#) – [G131](#) – [H44](#) – [H65](#)

Critère 11.2 Chaque étiquette associée à un champ de formulaire est-elle pertinente (hors cas particuliers) ?

- Test 11.2.1 : Chaque balise `<label>` permet-elle de connaître la fonction exacte du champ de formulaire auquel elle est associée ?
- Test 11.2.2 : Chaque attribut `title` permet-il de connaître la fonction exacte du champ de formulaire auquel il est associé ?
- Test 11.2.3 : Chaque étiquette implémentée via l'attribut WAI-ARIA `aria-label` permet-elle de connaître la fonction exacte du champ de formulaire auquel elle est associée ?
- Test 11.2.4 : Chaque passage de texte associé via l'attribut WAI-ARIA `aria-labelledby` permet-il de connaître la fonction exacte du champ de formulaire auquel il est associé ?
- Test 11.2.5 : Chaque champ de formulaire ayant un intitulé visible vérifie-t-il ces conditions (hors cas particuliers) ?
 - S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` du champ de formulaire contient au moins l'intitulé visible ;
 - S'il est présent, le passage de texte lié au champ de formulaire via un attribut WAI-ARIA `aria-labelledby` contient au moins l'intitulé visible ;
 - S'il est présent, le contenu de l'attribut `title` du champ de formulaire contient au moins l'intitulé visible ;
 - S'il est présent le contenu de la balise `<label>` associé au champ de formulaire contient au moins l'intitulé visible.
- Test 11.2.6 : Chaque bouton adjacent au champ de formulaire qui fournit une étiquette visible permet-il de connaître la fonction exacte du champ de formulaire auquel il est associé ?

Cas particuliers

Il existe une gestion de cas particuliers pour le test 11.2.5 lorsque :

- La ponctuation et les lettres majuscules sont présentes dans le texte de l'intitulé visible : elles peuvent être ignorées dans le nom accessible sans porter à conséquence ;
- Le texte de l'intitulé visible sert de symbole : le texte ne doit pas être interprété littéralement au niveau du nom accessible. Le nom doit exprimer la fonction véhiculée par le symbole (par exemple, "B" au niveau d'un éditeur de texte aura pour nom accessible "Mettre en gras", le signe ">" en fonction du contexte signifiera "Suivant" ou "Lancer la vidéo"). Le cas des symboles mathématiques fait cependant exception (voir la note ci-dessous).

Note : si l'étiquette visible représente une expression mathématique, les symboles mathématiques peuvent être repris littéralement pour servir d'étiquette au nom accessible (ex. : "A>B"). Il est laissé à l'utilisateur le soin d'opérer la correspondance entre l'expression et ce qu'il doit épeler compte tenu

de la connaissance qu'il a du fonctionnement de son logiciel de saisie vocale ("A plus grand que B" ou "A supérieur à B").

Ce cas particulier s'applique également au test 11.9.2.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.4.6 Headings and Labels (AA) / [2.4.6 En-têtes et étiquettes](#)
- 9.2.5.3 Label in Name (A) / [2.5.3 Étiquette dans le nom](#)
- 9.3.3.2 Labels or Instructions (A) / [3.3.2 Étiquettes ou instructions](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA6](#) – [ARIA9](#) – [ARIA14](#) – [ARIA16](#) – [G82](#) – [G131](#) – [H44](#) – [H65](#)

Critère 11.3 Dans chaque [formulaire](#), chaque [étiquette](#) associée à un [champ de formulaire](#) ayant la même fonction et répétée plusieurs fois dans une même page ou dans un [ensemble de pages](#) est-elle [cohérente](#) ?

- Test 11.3.1 : Chaque [étiquette](#) associée à un [champ de formulaire](#) ayant la même fonction et répétée plusieurs fois dans une même page est-elle [cohérente](#) ?
- Test 11.3.2 : Chaque [étiquette](#) associée à un [champ de formulaire](#) ayant la même fonction et répétée dans un ensemble de pages est-elle [cohérente](#) ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.3.2.4 Consistent Identification (AA) / [3.2.4 Identification cohérente](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F31](#)

Critère 11.4 Dans chaque [formulaire](#), chaque [étiquette de champ](#) et son [champ associé](#) sont-ils [accolés](#) (hors cas particuliers) ?

- Test 11.4.1 : Chaque [étiquette de champ](#) et son [champ](#) associé sont-ils [accolés](#) ?
- Test 11.4.2 : Chaque [étiquette accolée](#) à un [champ](#) (à l'exception des cases à cocher, bouton radio ou balises ayant un attribut WAI-ARIA `role="checkbox"`, `role="radio"` ou `role="switch"`), vérifie-t-elle ces conditions (hors cas particuliers) ?
 - L'étiquette est visuellement [accolée](#) immédiatement au-dessus ou à gauche du [champ de formulaire](#) lorsque le sens de lecture de la langue de l'étiquette est de gauche à droite ;
 - L'étiquette est visuellement [accolée](#) immédiatement au-dessus ou à droite du [champ de formulaire](#) lorsque le sens de lecture de la langue de l'étiquette est de droite à gauche.
- Test 11.4.3 : Chaque [étiquette accolée](#) à un [champ](#) de type `checkbox` ou `radio` ou à une balise ayant un attribut WAI-ARIA `role="checkbox"`, `role="radio"` ou `role="switch"`, vérifie-t-elle ces conditions (hors cas particuliers) ?

- L'étiquette est visuellement accolée immédiatement au-dessous ou à droite du champ de formulaire lorsque le sens de lecture de la langue de l'étiquette est de gauche à droite ;
- L'étiquette est visuellement accolée immédiatement au-dessous ou à gauche du champ de formulaire lorsque le sens de lecture de la langue de l'étiquette est de droite à gauche.

Cas particuliers

Les tests 11.4.2 et 11.4.3 seront considérés comme non applicables :

- Dans le cas où l'étiquette mélange une portion de texte qui se lit de droite à gauche avec une portion de texte qui se lit de gauche à droite ;
- Dans le cas où un formulaire contient des labels de plusieurs langues qui se liraient de droite à gauche et inversement. Par exemple, un formulaire de commande en arabe qui propose une liste de cases à cocher de produit en langue française ou mixant des produits en langue arabe ou en langue française ;
- Dans le cas où les champs de type `radio` ou `checkbox` et les balises ayant un attribut WAI-ARIA `role="checkbox"`, `role="radio"` ou `role="switch"` ne sont pas visuellement présentés sous forme de bouton radio ou de case à cocher ;
- Dans le cas où les champs seraient utilisés dans un contexte où il pourrait être légitime, du point de vue de l'expérience utilisateur, de placer les étiquettes de manière différente à celle requise dans les tests 11.4.2 et 11.4.3.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.3.3.2 Labels or Instructions (A) / 3.3.2 Étiquettes ou instructions

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : G162

Critère 11.5 Dans chaque formulaire, les champs de même nature sont-ils regroupés, si nécessaire ?

- Test 11.5.1 : Les champs de même nature vérifient-ils l'une de ces conditions, si nécessaire ?
 - Les champs de même nature sont regroupés dans une balise `<fieldset>` ;
 - Les champs de même nature sont regroupés dans une balise possédant un attribut WAI-ARIA `role="group"` ;
 - Les champs de même nature de type radio (`<input type="radio">`) ou balises possédant un attribut WAI-ARIA `role="radio"`) sont regroupés dans une balise possédant un attribut WAI-ARIA `role="radiogroup"` ou `role="group"`.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)
- 9.3.3.2 Labels or Instructions (A) / [3.3.2 Étiquettes ou instructions](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA17](#) – [H71](#)

Critère 11.6 Dans chaque formulaire, chaque regroupement de champs de même nature a-t-il une légende ?

- Test 11.6.1 : Chaque regroupement de [champs de même nature](#) possède-t-il une [légende](#) ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)
- 9.3.3.2 Labels or Instructions (A) / [3.3.2 Étiquettes ou instructions](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA17](#) – [H71](#)

Critère 11.7 Dans chaque formulaire, chaque légende associée à un regroupement de champs de même nature est-elle pertinente ?

- Test 11.7.1 : Chaque [légende](#) associée à un regroupement de [champs de même nature](#) est-elle pertinente ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)
- 9.3.3.2 Labels or Instructions (A) / [3.3.2 Étiquettes ou instructions](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA17](#) – [H71](#)

Critère 11.8 Dans chaque formulaire, les items de même nature d'une liste de choix sont-ils regroupés de manière pertinente ?

- Test 11.8.1 : Pour chaque balise `<select>`, les [items de même nature d'une liste de choix](#) sont-ils regroupés avec une balise `<optgroup>`, si nécessaire ?
- Test 11.8.2 : Dans chaque balise `<select>`, chaque balise `<optgroup>` possède-t-elle un attribut `label` ?
- Test 11.8.3 : Pour chaque balise `<optgroup>` ayant un attribut `label`, le contenu de l'attribut `label` est-il pertinent ?

Note technique

Il est possible d'utiliser une balise ayant un attribut WAI-ARIA `role="listbox"` en remplacement d'une balise `<select>`. En revanche, il est impossible de créer des groupes d'options via l'utilisation de

WAI-ARIA. De ce fait, une liste nécessitant un regroupement d'options structurée à l'aide d'une balise ayant un attribut WAI-ARIA `role="listbox"` sera considérée comme non conforme au critère 11.8.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [H85](#)

Critère 11.9 Dans chaque [formulaire](#), l'intitulé de chaque [bouton](#) est-il pertinent (hors cas particuliers) ?

- Test 11.9.1 : L'intitulé de chaque [bouton](#) vérifie-t-il ces conditions (hors cas particuliers) ?
 - S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` est pertinent ;
 - S'il est présent, le [passage de texte](#) lié au bouton via un attribut WAI-ARIA `aria-labelledby` est pertinent ;
 - S'il est présent, le contenu de l'attribut `value` d'une balise `<input>` de type `submit`, `reset` ou `button` est pertinent ;
 - S'il est présent, le contenu de la balise `<button>` est pertinent ;
 - S'il est présent, le contenu de l'attribut `alt` d'une balise `<input>` de type `image` est pertinent ;
 - S'il est présent, le contenu de l'attribut `title` est pertinent.
- Test 11.9.2 : Chaque [bouton](#) affichant un [intitulé visible](#) vérifie-t-il ces conditions (hors cas particuliers) ?
 - S'il est présent, le contenu de l'attribut WAI-ARIA `aria-label` contient au moins l'[intitulé visible](#) ;
 - S'il est présent, le [passage de texte](#) lié au bouton via un attribut WAI-ARIA `aria-labelledby` contient au moins l'[intitulé visible](#) ;
 - S'il est présent, le contenu de l'attribut `value` d'une balise `<input>` de type `submit`, `reset` ou `button` contient au moins l'[intitulé visible](#) ;
 - S'il est présent, le contenu de la balise `<button>` contient au moins l'[intitulé visible](#) ;
 - S'il est présent, le contenu de l'attribut `alt` d'une balise `<input>` de type `image` contient au moins l'[intitulé visible](#) ;
 - S'il est présent, le contenu de l'attribut `title` contient au moins l'[intitulé visible](#).

Cas particulier

Pour le test 11.9.2, voir cas particuliers critère 11.2.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.5.3 Label in Name (A) / [2.5.3 Étiquette dans le nom](#)
- 9.4.1.2 Name, Role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA6](#) – [ARIA9](#) – [ARIA14](#) – [ARIA16](#) – [H36](#) – [H91](#)

Critère 11.10 Dans chaque [formulaire](#), le [contrôle de saisie](#) est-il utilisé de manière pertinente (hors cas particuliers) ?

- Test 11.10.1 : Les [indications du caractère obligatoire](#) de la saisie des champs vérifient-elles une de ces conditions (hors cas particuliers) ?
 - Une [indication de champ obligatoire](#) est visible et permet d'identifier nommément le champ concerné préalablement à la validation du formulaire ;
 - Le champ obligatoire dispose de l'attribut `aria-required="true"` **ou** `required` préalablement à la validation du formulaire.
- Test 11.10.2 : Les champs obligatoires ayant l'attribut `aria-required="true"` **ou** `required` vérifient-ils une de ces conditions ?
 - Une [indication de champ obligatoire](#) est visible et située dans l'étiquette associée au champ préalablement à la validation du formulaire ;
 - Une [indication de champ obligatoire](#) est visible et située dans le [passage de texte](#) associé au champ préalablement à la validation du formulaire.
- Test 11.10.3 : Les messages d'erreur indiquant l'absence de saisie d'un champ obligatoire vérifient-ils une de ces conditions ?
 - Le message d'erreur indiquant l'absence de saisie d'un champ obligatoire est visible et permet d'identifier nommément le champ concerné ;
 - Le champ obligatoire dispose de l'attribut `aria-invalid="true"`.
- Test 11.10.4 : Les champs obligatoires ayant l'attribut `aria-invalid="true"` vérifient-ils une de ces conditions ?
 - Le message d'erreur indiquant le caractère invalide de la saisie est visible et situé dans l'étiquette associée au champ ;
 - Le message d'erreur indiquant le caractère invalide de la saisie est visible et situé dans le [passage de texte](#) associé au champ.

- Test 11.10.5 : Les instructions et indications du type de données et/ou de format obligatoires vérifient-elles une de ces conditions ?
 - Une instruction ou une indication du type de données et/ou de format obligatoire est visible et permet d'identifier nommément le champ concerné préalablement à la validation du formulaire ;
 - Une instruction ou une indication du type de données et/ou de format obligatoire est visible dans l'étiquette ou le [passage de texte](#) associé au champ préalablement à la validation du formulaire.
- Test 11.10.6 : Les messages d'erreurs fournissant une instruction ou une indication du type de données et/ou de format obligatoire des champs vérifient-ils une de ces conditions ?
 - Le message d'erreur fournissant une instruction ou une indication du type de données et/ou de format obligatoires est visible et identifie le champ concerné ;
 - Le champ dispose de l'attribut `aria-invalid="true"`.
- Test 11.10.7 : Les champs ayant l'attribut `aria-invalid="true"` dont la saisie requiert un type de données et/ou de format obligatoires vérifient-ils une de ces conditions ?
 - Une instruction ou une indication du type de données et/ou de format obligatoire est visible et située dans la balise `<label>` associée au champ ;
 - Une instruction ou une indication du type de données et/ou de format obligatoire est visible et située dans le [passage de texte](#) associé au champ.

Cas particuliers

Le test 11.10.1 et le test 11.10.2 seront considérés comme non applicables lorsque le formulaire comporte un seul [champ de formulaire](#) ou qu'il indique les champs optionnels de manière :

- Visible ;
- Dans la balise `<label>` ou dans la [légende](#) associée au champ.

Dans le cas où l'ensemble des champs d'un formulaire sont obligatoires, les tests 11.10.1 et 11.10.2 restent applicables.

Notes techniques

Dans un long formulaire dont la majorité des champs sont obligatoires, on pourrait constater que ce sont les quelques champs restés facultatifs qui sont explicitement signalés comme tels. Dans ce cas, il faudrait s'assurer que :

- Un message précise visuellement en haut de formulaire que "tous les champs sont obligatoires sauf ceux indiqués comme étant facultatifs" ;

- Une mention "facultatif" est présente visuellement dans le libellé des champs facultatifs ou dans la légende d'un groupe de champs facultatifs ;
- Un attribut `required` ou `aria-required="true"` reste associé à chaque champ qui n'est pas concerné par ce caractère facultatif.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.3.3.1 Error Identification (A) / [3.3.1 Identification des erreurs](#)
- 9.3.3.2 Labels or Instructions (A) / [3.3.2 Étiquettes ou instructions](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA1](#) – [ARIA2](#) – [ARIA6](#) – [ARIA9](#) – [ARIA16](#) – [ARIA21](#) – [F81](#) – [G83](#) – [G84](#) – [G85](#) – [G89](#) – [G184](#) – [H44](#) – [H81](#) – [H89](#) – [H90](#) – [SCR18](#) – [SCR32](#)

Critère 11.11 Dans chaque [formulaire](#), le [contrôle de saisie](#) est-il accompagné, si nécessaire, de suggestions facilitant la correction des erreurs de saisie ?

- Test 11.11.1 : Pour chaque erreur de saisie, les types et les formats de données sont-ils suggérés, si nécessaire ?
- Test 11.11.2 : Pour chaque erreur de saisie, des exemples de valeurs attendues sont-ils suggérés, si nécessaire ?

Note technique

Certains types de contrôles en HTML5 proposent des messages d'aide à la saisie automatique : par exemple le type `email` affiche un message du type « veuillez saisir une adresse e-mail valide » dans le cas où l'adresse e-mail saisie ne correspond pas au format attendu. Ces messages sont personnalisables via l'API Constraint Validation, ce qui permet de personnaliser les messages d'erreur et de valider le critère. L'attribut `pattern` permet d'effectuer automatiquement des contrôles de format (via des expressions régulières) et affiche un message d'aide personnalisable via l'attribut `title` : ce dispositif valide également le critère.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.3.3.3 Error Suggestion (AA) / [3.3.3 Suggestion après une erreur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G84](#) – [G85](#) – [G89](#) – [G177](#) – [H89](#)

Critère 11.12 Pour chaque [formulaire](#) qui modifie ou supprime des données, ou qui transmet des réponses à un test ou à un examen, ou dont la validation a des conséquences financières ou juridiques, les données saisies peuvent-elles être modifiées, mises à jour ou récupérées par l'utilisateur ?

- Test 11.12.1 : Pour chaque formulaire qui modifie ou supprime des données, ou qui transmet des réponses à un test ou un examen, ou dont la validation a des conséquences financières ou juridiques, la saisie des données vérifie-t-elle une de ces conditions ?

- L'utilisateur peut [modifier ou annuler les données et les actions effectuées](#) sur ces données après la validation du formulaire ;
 - L'utilisateur peut vérifier et corriger les données avant la validation d'un formulaire en plusieurs étapes ;
 - Un mécanisme de confirmation explicite, via une case à cocher (balise `<input>` de type `checkbox` ou balise ayant un attribut WAI-ARIA `role="checkbox"`) ou une étape supplémentaire, est présent.
- Test 11.12.2 : Chaque formulaire dont la validation modifie ou supprime des données à caractère financier, juridique ou personnel vérifie-t-il une de ces conditions ?
 - Un mécanisme permet de récupérer les données supprimées ou modifiées par l'utilisateur ;
 - Un mécanisme de demande de confirmation explicite de la suppression ou de la modification, via un [champ de formulaire](#) ou une étape supplémentaire, est proposé.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.3.3.4 Error Prevention (Legal, Financial, Data) (AA) / [3.3.4 Prévention des erreurs \(juridiques, financières, de données\)](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G98](#) – [G99](#) – [G155](#) – [G164](#) – [G168](#)

Critère 11.13 La finalité d'un champ de saisie peut-elle être déduite pour faciliter le remplissage automatique des champs avec les données de l'utilisateur ?

- Test 11.13.1 : Chaque [champ de formulaire](#) dont l'objet se rapporte à une information concernant l'utilisateur vérifie-t-il ces conditions ?
 - Le [champ de formulaire](#) possède un attribut `autocomplete` ;
 - L'attribut `autocomplete` est pourvu d'une valeur présente dans la [liste des valeurs possibles pour l'attribut `autocomplete`](#) associés à un [champ de formulaire](#) ;
 - La valeur indiquée pour l'attribut `autocomplete` est pertinente au regard du type d'information attendu.

Notes techniques

La [liste des valeurs possibles pour l'attribut `autocomplete`](#) repose sur la liste des valeurs présentes dans la spécification WCAG2.1 qui reprend elle-même la liste des valeurs de type "field name" de la spécification HTML5.2. Le critère WCAG demande à ce que l'une de ces valeurs soit présente pour qualifier un champ de saisie concernant l'utilisateur.

Ce que le critère WCAG laisse implicite, ce sont les différentes règles de construction possibles pour obtenir une valeur (simple ou composée) pour l'attribut `autocomplete`. C'est cependant l'affaire du

développeur de fournir à l'attribut `autocomplete` une valeur ou un ensemble de valeurs valides au regard des exigences de l'algorithme fourni par la spécification HTML5.2. Ainsi, un attribut `autocomplete` ne peut contenir qu'une seule valeur de type "field name", comme "name" ou "street-address". On peut avoir également un ensemble composé de différentes valeurs comme, par exemple, `autocomplete="shipping name" OU autocomplete="section-software shipping street-address" : "section-software"` renvoie à une valeur de type "scope" et "shipping" à une valeur de type "hint set", mais toujours une seule valeur de type "field name".

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.5 Identify Input Purpose (AA) / [1.3.5 Identifier la finalité de la saisie](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [H98](#)

2.2.12.12. Navigation

Critère 12.1 Chaque [ensemble de pages](#) dispose-t-il de deux [systèmes de navigation](#) différents, au moins (hors cas particuliers) ?

- Test 12.1.1 : Chaque [ensemble de pages](#) vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - Un [menu de navigation](#) et un [plan du site](#) sont présents ;
 - Un [menu de navigation](#) et un [moteur de recherche](#) sont présents ;
 - Un [moteur de recherche](#) et un [plan du site](#) sont présents.

Cas particuliers

Il existe une gestion de cas particulier lorsque le site web est constitué d'une seule page ou d'un nombre très limité de pages (cf. note). Dans ce cas-là, le critère est non applicable.

Le critère est également non applicable pour les pages d'un ensemble de pages qui sont le résultat ou une partie d'un processus (un processus de paiement ou de prise de commande, par exemple).

Note : l'appréciation d'un nombre très limité de pages devrait être réservé à un site dont l'ensemble des pages sont atteignables depuis la page d'accueil.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.4.5 Multiple Ways (AA) / [2.4.5 Accès multiples](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G63](#) – [G64](#) – [G161](#)

Critère 12.2 Dans chaque [ensemble de pages](#), le [menu et les barres de navigation](#) sont-ils toujours à la même place (hors cas particuliers) ?

- Test 12.2.1 : Dans chaque [ensemble de pages](#), chaque page disposant d'un [menu et les barres de navigation](#) vérifie-t-elle ces conditions (hors cas particuliers) ?
 - Le [menu et les barres de navigation](#) sont toujours à la même place dans la présentation ;
 - Le [menu et les barres de navigation](#) se présentent toujours dans le même ordre relatif dans le code source.

Cas particuliers

Il existe une gestion de cas particuliers lorsque :

- La page est la page d'accueil ;
- Le site web est constitué d'une seule page ;
- Le changement fait suite à une modification initiée par l'utilisateur.

Dans ces situations, le critère est non applicable.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.3.2.3 Consistent Navigation (AA) / [3.2.3 Navigation cohérente](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F66](#) – [G61](#)

Critère 12.3 La [page « plan du site »](#) est-elle pertinente ?

- Test 12.3.1 : La [page « plan du site »](#) est-elle représentative de l'architecture générale du site ?
- Test 12.3.2 : Les liens du [plan du site](#) sont-ils fonctionnels ?
- Test 12.3.3 : Les liens du [plan du site](#) renvoient-ils bien vers les pages indiquées par l'intitulé ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.4.5 Multiple Ways (AA) / [2.4.5 Accès multiples](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G63](#)

Critère 12.4 Dans chaque [ensemble de pages](#), la [page « plan du site »](#) est-elle accessible à partir d'une fonctionnalité identique ?

- Test 12.4.1 : Dans chaque [ensemble de pages](#), la [page « plan du site »](#) est-elle accessible à partir d'une fonctionnalité identique ?
- Test 12.4.2 : Dans chaque [ensemble de pages](#), la fonctionnalité vers la [page « plan du site »](#) est-elle située à la même place dans la présentation ?

- Test 12.4.3 : Dans chaque [ensemble de pages](#), la fonctionnalité vers la [page « plan du site »](#) se présente-t-elle toujours dans le même ordre relatif dans le code source ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.4.5 Multiple Ways (AA) / [2.4.5 Accès multiples](#)
- 9.3.2.3 Consistent Navigation (AA) / [3.2.3 Navigation cohérente](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G61](#) – [G63](#)

Critère 12.5 Dans chaque [ensemble de pages](#), le [moteur de recherche](#) est-il atteignable de manière identique ?

- Test 12.5.1 : Dans chaque [ensemble de pages](#), le [moteur de recherche](#) est-il accessible à partir d'une fonctionnalité identique ?
- Test 12.5.2 : Dans chaque [ensemble de pages](#), la fonctionnalité vers le [moteur de recherche](#) est-elle située à la même place dans la présentation ?
- Test 12.5.3 : Dans chaque [ensemble de pages](#), la fonctionnalité vers le [moteur de recherche](#) se présente-t-elle toujours dans le même ordre relatif dans le code source ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.3.2.3 Consistent Navigation (AA) / [3.2.3 Navigation cohérente](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F66](#) – [G61](#)

Critère 12.6 Les zones de regroupement de contenus présentes dans plusieurs pages web ([zones d'en-tête](#), de [navigation principale](#), de [contenu principal](#), de [pied de page](#) et de [moteur de recherche](#)) peuvent-elles être atteintes ou évitées ?

- Test 12.6.1 : Dans chaque page web où elles sont présentes, la zone d'[en-tête](#), de [navigation principale](#), de [contenu principal](#), de [pied de page](#) et de [moteur de recherche](#) respectent-elles au moins une de ces conditions ?
 - La zone possède un rôle WAI-ARIA de type [landmark](#) correspondant à sa nature ;
 - La zone possède un titre dont le contenu permet de comprendre la nature du contenu de la zone ;
 - La zone peut être masquée par le biais d'un bouton précédent directement la zone dans l'ordre du code source ;
 - La zone peut être évitée par le biais d'un [lien d'évitement](#) précédent directement la zone dans l'ordre du code source ;

- La zone peut être atteinte par le biais d'un [lien d'accès rapide](#) visible ou, à défaut, visible à la prise de focus.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)
- 9.2.4.1 Bypass Blocks (A) / [2.4.1 Contourner des blocs](#)
- 9.4.1.2 Name, Role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [ARIA4](#) – [ARIA11](#) – [G115](#) – [H69](#)

Critère 12.7 Dans chaque page web, un [lien d'évitement ou d'accès rapide](#) à la [zone de contenu principal](#) est-il présent (hors cas particuliers) ?

- Test 12.7.1 : Dans chaque page web, un lien permet-il d'éviter la [zone de contenu principal](#) ou d'y accéder (hors cas particuliers) ?
- Test 12.7.2 : Dans chaque ensemble de pages, le [lien d'évitement ou d'accès rapide](#) à la [zone de contenu principal](#) vérifie-t-il ces conditions (hors cas particuliers) ?
 - Le lien est situé à la même place dans la présentation ;
 - Le lien se présente toujours dans le même ordre relatif dans le code source ;
 - Le lien est visible ou, à défaut, visible à la prise de focus ;
 - Le lien est fonctionnel.

Cas particuliers

Il existe une gestion de cas particuliers lorsque le site web est constitué d'une seule page.

Dans ce cas de figure, l'obligation de la présence d'un lien d'accès rapide est liée au contexte de la page : présence ou absence de navigation ou de contenus additionnels, par exemple. Le critère peut être considéré comme non applicable lorsqu'il est avéré qu'un lien d'accès rapide est inutile.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.4.1 Bypass Blocks (A) / [2.4.1 Contourner des blocs](#)
- 9.2.4.3 Focus Order (A) / [2.4.3 Parcours du focus](#)
- 9.3.2.3 Consistent Navigation (AA) / [3.2.3 Navigation cohérente](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F66](#) – [G1](#) – [G59](#) – [G123](#) – [G124](#) – [SCR28](#)

Critère 12.8 Dans chaque page web, l'[ordre de tabulation](#) est-il [cohérent](#) ?

- Test 12.8.1 : Dans chaque page web, l'[ordre de tabulation](#) dans le contenu est-il [cohérent](#) ?

- Test 12.8.2 : Pour chaque [script](#) qui met à jour ou insère un contenu, l'[ordre de tabulation](#) reste-t-il [cohérent](#) ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.4.3 Focus Order (A) / [2.4.3 Parcours du focus](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [C27](#) – [F44](#) – [F85](#) – [G59](#) – [H4](#) – [SCR26](#) – [SCR27](#) – [SCR37](#)

Critère 12.9 Dans chaque page web, la navigation ne doit pas contenir de piège au clavier. Cette règle est-elle respectée ?

- Test 12.9.1 : Dans chaque page web, chaque [élément recevant le focus](#) vérifie-t-il une de ces conditions ?
 - Il est possible d'atteindre l'élément suivant ou précédent pouvant recevoir le focus avec la touche de tabulation ;
 - L'utilisateur est informé d'un mécanisme fonctionnel permettant d'atteindre au clavier l'élément suivant ou précédent pouvant recevoir le focus.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.1.1 Keyboard (A) / [2.1.1 Clavier](#)
- 9.2.1.2 No Keyboard Trap (A) / [2.1.2 Pas de piège au clavier](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F10](#) – [G21](#) – [H91](#)

Critère 12.10 Dans chaque page web, les [raccourcis clavier](#) n'utilisant qu'une seule touche (lettre minuscule ou majuscule, ponctuation, chiffre ou symbole) sont-ils contrôlables par l'utilisateur ?

- Test 12.10.1 : Dans chaque page web, chaque [raccourci clavier](#) n'utilisant qu'une seule touche (lettre minuscule ou majuscule, ponctuation, chiffre ou symbole) vérifie-t-il l'une de ces conditions ?
 - Un mécanisme est disponible pour désactiver le [raccourci clavier](#) ;
 - Un mécanisme est disponible pour configurer la touche de [raccourci clavier](#) au moyen des touches de modification (Ctrl, Alt, Maj, etc.) ;
 - Dans le cas d'un [composant d'interface](#) utilisateur, le [raccourci clavier](#) qui lui est associé ne peut être activé que si le focus clavier est sur ce composant.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.1.4 Character Key Shortcuts (A) / [2.1.4 Raccourcis clavier utilisant des caractères](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F99](#) – [G217](#)

Critère 12.11 Dans chaque page web, les contenus additionnels apparaissant au survol, à la prise de focus ou à l'activation d'un [composant d'interface](#) sont-ils si nécessaire atteignables au clavier ?

- Test 12.11.1 : Dans chaque page web, les contenus additionnels apparaissant au survol, à la prise de focus ou à l'activation d'un [composant d'interface](#) sont-ils si nécessaire atteignables au clavier ?

Note technique

Ce critère adresse les situations où un contenu additionnel contient des [composants d'interface](#) avec lesquels il doit être possible d'interagir au clavier. Par exemple, une infobulle personnalisée qui propose un lien dans son contenu.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.1.1 Keyboard (A) / [2.1.1 Clavier](#)

2.2.13.13. Consultation

Critère 13.1 Pour chaque page web, l'utilisateur a-t-il le contrôle de chaque limite de temps modifiant le contenu (hors cas particuliers) ?

- Test 13.1.1 : Pour chaque page web, chaque [procédé de rafraîchissement](#) (balise `<object>`, balise `<embed>`, balise `<svg>`, balise `<canvas>`, balise `<meta>`) vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - L'utilisateur peut arrêter ou relancer le rafraîchissement ;
 - L'utilisateur peut augmenter la limite de temps entre deux rafraîchissements de dix fois, au moins ;
 - L'utilisateur est averti de l'imminence du rafraîchissement et dispose de vingt secondes, au moins, pour augmenter la limite de temps avant le prochain rafraîchissement ;
 - La limite de temps entre deux rafraîchissements est de vingt heures, au moins.
- Test 13.1.2 : Pour chaque page web, chaque procédé de [redirection](#) effectué via une balise `<meta>` est-il immédiat (hors cas particuliers) ?
- Test 13.1.3 : Pour chaque page web, chaque procédé de [redirection](#) effectué via un [script](#) vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - L'utilisateur peut arrêter ou relancer la redirection ;
 - L'utilisateur peut augmenter la limite de temps avant la redirection de dix fois, au moins ;

- L'utilisateur est averti de l'imminence de la redirection et dispose de vingt secondes, au moins, pour augmenter la limite de temps avant la prochaine redirection ;
- La limite de temps avant la redirection est de vingt heures, au moins.
- Test 13.1.4 : Pour chaque page web, chaque procédé limitant le temps d'une session vérifie-t-il une de ces conditions (hors cas particuliers) ?
 - L'utilisateur peut supprimer la limite de temps ;
 - L'utilisateur peut augmenter la limite de temps ;
 - La limite de temps avant la fin de la session est de vingt heures au moins.

Cas particuliers

Il existe une gestion de cas particuliers lorsque la limite de temps est essentielle, notamment lorsqu'elle ne pourrait pas être supprimée sans changer fondamentalement le contenu ou les fonctionnalités liées au contenu.

Dans ces situations, le critère est non applicable. Par exemple, le rafraîchissement d'un flux RSS dans une page n'est pas une limite de temps essentielle ; le critère est applicable. En revanche, une redirection automatique qui amène vers la nouvelle version d'une page à partir d'une URL obsolète est essentielle ; le critère est non applicable.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.2.1 Timing Adjustable (A) / [2.2.1 Réglage du délai](#)
- 9.2.2.2 Pause, Stop, Hide (A) / [2.2.2 Mettre en pause, arrêter, masquer](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F40](#) – [F41](#) – [F58](#) – [F61](#) – [G75](#) – [G76](#) – [G110](#) – [G133](#) – [G180](#) – [G186](#) – [G198](#) – [H76](#) – [SCR1](#) – [SCR16](#) – [SCR36](#) – [SVR1](#)

Critère 13.2 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 ?

- Test 13.2.1 : 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 ?

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.3.2.1 On focus (A) / [3.2.1 Au focus](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F55](#) – [G107](#)

Critère 13.3 Dans chaque page web, chaque document bureautique en téléchargement possède-t-il, si nécessaire, une version accessible (hors cas particuliers) ?

- Test 13.3.1 : Dans chaque page web, chaque fonctionnalité de téléchargement d'un document bureautique vérifie-t-elle une de ces conditions ?
 - Le document en téléchargement est compatible avec l'accessibilité ;
 - Il en existe une version alternative en téléchargement compatible avec l'accessibilité ;
 - Il en existe une version alternative au format HTML compatible avec l'accessibilité.

Cas particuliers

Il existe une gestion de cas particuliers :

- Pour les personnes de droit privé mentionnées aux 2° à 4° du I de l'article 47 de la loi du 11 février 2005 : si les fichiers bureautiques (ex : PDF, documents Microsoft ou LibreOffice, etc.) ont été publiés avant le 23 septembre 2018 (sauf si ce sont des documents nécessaires pour accomplir une démarche administrative relevant des tâches effectuées par l'organisme concerné), ils sont exemptés de l'obligation d'accessibilité.

Dans cette situation, le critère est non applicable.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)
- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)
- 9.1.3.2 Meaningful Sequence (A) / [1.3.2 Ordre séquentiel logique](#)
- 9.2.4.1 Bypass Blocks (A) / [2.4.1 Contourner des blocs](#)
- 9.2.4.3 Focus Order (A) / [2.4.3 Parcours du focus](#)
- 9.3.1.1 Language of Page (A) / [3.1.1 Langue de la page](#)
- 9.4.1.2 Name, Role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F15](#) – [G10](#) – [G135](#)

Critère 13.4 Pour chaque document bureautique ayant une version accessible, cette version offre-t-elle la même information ?

- Test 13.4.1 : Chaque document bureautique ayant une version accessible vérifie-t-il une de ces conditions ?
 - La version compatible avec l'accessibilité offre la même information ;
 - La version alternative au format HTML est pertinente et offre la même information.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)
- 9.1.3.1 Info and Relationships (A) / [1.3.1 Information et relations](#)
- 9.1.3.2 Meaningful Sequence (A) / [1.3.2 Ordre séquentiel logique](#)

- 9.2.4.1 Bypass Blocks (A) / [2.4.1 Contourner des blocs](#)
- 9.2.4.3 Focus Order (A) / [2.4.3 Parcours du focus](#)
- 9.3.1.1 Language of Page (A) / [3.1.1 Langue de la page](#)
- 9.4.1.2 Name, Role, Value (A) / [4.1.2 Nom, rôle et valeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F15](#) – [G10](#) – [G135](#)

Critère 13.5 Dans chaque page web, chaque contenu cryptique (art ASCII, émoticône, syntaxe cryptique) a-t-il une alternative ?

- Test 13.5.1 : Dans chaque page web, chaque contenu cryptique (art ASCII, émoticône, syntaxe cryptique) vérifie-t-il une de ces conditions ?
 - Un attribut title est disponible ;
 - Une définition est donnée par le contexte adjacent.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F70](#) – [F71](#) – [G135](#) – [H86](#)

Critère 13.6 Dans chaque page web, pour chaque contenu cryptique (art ASCII, émoticône, syntaxe cryptique) ayant une alternative, cette alternative est-elle pertinente ?

- Test 13.6.1 : Dans chaque page web, chaque contenu cryptique (art ASCII, émoticône, syntaxe cryptique) vérifie-t-il une de ces conditions ?
 - Le contenu de l'attribut `title` est pertinent ;
 - La définition donnée par le contexte adjacent est pertinente.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.1.1 Non-text Content (A) / [1.1.1 Contenu non textuel](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F71](#) – [F72](#) – [H86](#)

Critère 13.7 Dans chaque page web, [les changements brusques de luminosité ou les effets de flash](#) sont-ils correctement utilisés ?

- Test 13.7.1 : Dans chaque page web, chaque image ou élément multimédia (balise `<video>`, balise ``, balise `<svg>`, balise `<canvas>`, balise `<embed>` ou balise `<object>`) qui provoque un changement brusque de luminosité ou un effet de flash vérifie-t-il une de ces conditions ?
 - La fréquence de l'effet est inférieure à 3 par seconde ;

- La surface totale cumulée des effets est inférieure ou égale à 21824 pixels.
- Test 13.7.2 : Dans chaque page web, chaque script qui provoque [un changement brusque de luminosité ou un effet de flash](#) vérifie-t-il une de ces conditions ?
 - La fréquence de l'effet est inférieure à 3 par seconde ;
 - La surface totale cumulée des effets est inférieure ou égale à 21824 pixels.
- Test 13.7.3 : Dans chaque page web, chaque mise en forme CSS qui provoque [un changement brusque de luminosité ou un effet de flash](#) vérifie-t-il une de ces conditions ?
 - La fréquence de l'effet est inférieure à 3 par seconde ;
 - La surface totale cumulée des effets est inférieure ou égale à 21824 pixels.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.3.1 Three Flashes or Below Threshold (A) / [2.3.1 Pas plus de trois flashes ou sous le seuil critique](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G15](#) – [G19](#) – [G176](#)

Critère 13.8 Dans chaque page web, chaque contenu en mouvement ou clignotant est-il contrôlable par l'utilisateur ?

- Test 13.8.1 : Dans chaque page web, chaque contenu en mouvement déclenché automatiquement, vérifie-t-il une de ces conditions ?
 - La durée du mouvement est inférieure ou égale à 5 secondes ;
 - L'utilisateur peut arrêter et relancer le mouvement ;
 - L'utilisateur peut afficher et masquer le contenu en mouvement ;
 - L'utilisateur peut afficher la totalité de l'information sans le mouvement.
- Test 13.8.2 : Dans chaque page web, chaque contenu clignotant déclenché automatiquement, vérifie-t-il une de ces conditions ?
 - La durée du clignotement est inférieure ou égale à 5 secondes ;
 - L'utilisateur peut arrêter et relancer le clignotement ;
 - L'utilisateur peut afficher et masquer le contenu clignotant ;
 - L'utilisateur peut afficher la totalité de l'information sans le clignotement.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.2.1 Timing Adjustable (A) / [2.2.1 Réglage du délai](#)
- 9.2.2.2 Pause, Stop, Hide (A) / [2.2.2 Mettre en pause, arrêter, masquer](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [F4](#) – [F7](#) – [F16](#) – [F47](#) – [F50](#) – [G4](#) – [G11](#) – [G152](#) – [G186](#) – [G187](#) – [G191](#) – [SCR22](#) – [SCR33](#) – [SCR36](#) – [SM11](#) – [SM12](#)

Critère 13.9 Dans chaque page web, le contenu proposé est-il consultable quelle que soit l'orientation de l'écran (portrait ou paysage) (hors cas particuliers) ?

- Test 13.9.1 : Dans chaque page web, chaque contenu vérifie-t-il ces conditions (hors cas particuliers) ?
 - La consultation est possible quel que soit le mode d'orientation de l'écran ;
 - Le contenu proposé reste le même quel que soit le mode d'orientation de l'écran utilisé même si sa présentation et le moyen d'y accéder peut différer.

Cas particuliers

Il existe des interfaces pour lesquelles l'orientation du périphérique est essentielle à leur utilisation.

Dans ces situations, le critère est non applicable. Il peut s'agir d'interfaces de jeu, de piano, de dépôt de chèques bancaires, etc.

Si l'interface est le seul moyen d'accéder au service proposé, une alternative devrait être mise en place pour pallier cette carence.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.1.3.4 Orientation (AA) / [1.3.4 Orientation](#)

Critère 13.10 Dans chaque page web, les fonctionnalités utilisables ou disponibles au moyen d'un [geste complexe](#) peuvent-elles être également disponibles au moyen d'un [geste simple](#) (hors cas particuliers) ?

- Test 13.10.1 : Dans chaque page web, chaque fonctionnalité utilisable ou disponible suite à un contact multipoint est-elle également utilisable ou disponible suite à un contact en un point unique de l'écran (hors cas particuliers).
- Test 13.10.2 : Dans chaque page web, chaque fonctionnalité utilisable ou disponible suite à un geste basé sur le suivi d'une trajectoire sur l'écran est-elle également utilisable ou disponible suite à un contact en un point unique de l'écran (hors cas particuliers).

Cas particuliers

Il existe une gestion de cas particuliers dans deux types de situation :

- Le critère ne s'applique qu'à des fonctionnalités mises en place par l'auteur du site. Il ne concerne donc pas les gestes requis par l'agent utilisateur ou le système d'exploitation ;
- Le critère ne s'applique pas aux fonctionnalités dont la réalisation d'un geste complexe est essentielle (exécuter le tracé d'une signature, par exemple).

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.5.1 Pointer Gestures (A) / [2.5.1 Gestes pour le contrôle du pointeur](#)

Technique(s) suffisante(s) et/ou échec(s) WCAG 2.1 : [G215](#) – [G216](#)

Critère 13.11 Dans chaque page web, les actions déclenchées au moyen d'un dispositif de pointage sur un point unique de l'écran peuvent-elles faire l'objet d'une annulation (hors cas particuliers) ?

- Test 13.11.1 : Dans chaque page web, les actions déclenchées au moyen d'un dispositif de pointage sur un point unique de l'écran vérifient-elles l'une de ces conditions (hors cas particuliers) ?
 - L'action est déclenchée au moment où le dispositif de pointage est [relâché ou relevé](#) ;
 - L'action est déclenchée au moment où le dispositif de pointage est [pressé ou posé](#) puis annulée lorsque le dispositif de pointage est [relâché ou relevé](#) ;
 - Un mécanisme est disponible pour abandonner (avant achèvement de l'action) ou annuler (après achèvement) l'exécution de l'action.

Cas particulier

Il existe une gestion de cas particulier lorsque la fonctionnalité nécessite que le comportement attendu soit réalisé lors d'un événement descendant, par exemple, un émulateur de clavier dont les touches doivent s'activer à la pression comme sur un clavier physique. Dans ces situations, le critère est non applicable.

Notes techniques

Deux exemples de mécanisme mis en place pour annuler ou abandonner une action déclenchée au moyen d'un dispositif de pointage sur un point unique de l'écran :

- Une fenêtre modale permettant d'annuler l'action après son achèvement ;
- Pour une fonction de glisser/déposer, le fait d'abandonner l'action si l'utilisateur relâche l'élément en dehors de la zone cible.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.5.2 Pointer Cancellation (A) / [2.5.2 Annulation de l'action du pointeur](#)

Critère 13.12 Dans chaque page web, les fonctionnalités qui impliquent un mouvement de l'appareil ou vers l'appareil peuvent-elles être satisfaites de manière alternative (hors cas particuliers) ?

- Test 13.12.1 : Dans chaque page web, les fonctionnalités disponibles en bougeant l'appareil peuvent-elles être accomplies avec des [composants d'interface](#) utilisateur (hors cas particuliers) ?
- Test 13.12.2 : Dans chaque page web, les fonctionnalités disponibles en faisant un geste en direction de l'appareil peuvent-elles être accomplies avec des [composants d'interface](#) utilisateur (hors cas particuliers) ?
- Test 13.12.3 : L'utilisateur a-t-il la possibilité de désactiver la détection du mouvement pour éviter un déclenchement accidentel de la fonctionnalité (hors cas particuliers) ?

Cas particuliers

Il existe une gestion de cas particulier lorsque :

- Le mouvement est essentiel à l'accomplissement de la fonctionnalité (ex. podomètre) ;
- La détection du mouvement est utilisée pour contrôler une fonctionnalité au travers d'une interface compatible avec l'accessibilité.

Références

EN 301 549 V2.1.2 / WCAG 2.1

- 9.2.5.4 Motion Actuation (A) / [2.5.4 Activation par le mouvement](#)

2.3 RGAA – Glossaire

2.3.1 A

Accessible et activable par le clavier et tout dispositif de pointage

- Un composant d'interface (lien, bouton...) est accessible au clavier et par tout dispositif de pointage lorsque l'utilisateur peut prendre, indifféremment, le focus par un pointeur ou la touche tabulation ;
- Un composant d'interface (lien, bouton...) est activable au clavier et par tout dispositif de pointage lorsque l'utilisateur peut enclencher, indifféremment, l'action prévue par le composant d'interface par une pression du pointeur ou la touche entrée du clavier ;
- Attention : pour certains composants d'interface comme les *sliders* (bouton coulissant ou rotatif...), il n'est pas possible de contrôler le composant par la seule touche d'entrée. Dans ces situations, d'autres touches (comme les touches de direction) peuvent être utilisées. En

particulier pour les éléments ayant un rôle WAI-ARIA correspondant à un motif de conception il est recommandé de considérer le document [WAI-ARIA 1.1 Authoring Practices](#) lors de leur implémentation.

Dans le référentiel, l'expression « contrôlable par le clavier et tout dispositif de pointage » se rapporte également à la présente définition.

Note importante : le recours à certaines technologies peut rendre la gestion du focus trop complexe ou trop instable pour ne reposer que sur la tabulation, les touches de direction et la touche entrée. Dans ce cas, la mise à disposition de raccourcis clavier peut être la seule solution pour rendre le composant utilisable.

Le critère ne peut être considéré comme conforme qu'à la condition que les raccourcis clavier utilisés soient correctement documentés, qu'ils soient fonctionnels et qu'ils respectent le [critère 12.10](#).

| Accolés (étiquette et champ accolés)

Il faut que l'étiquette et son champ soient visuellement proches de manière à ce que la relation entre les deux ne puisse pas prêter à confusion.

| Alternative (à script)

Texte ou procédé associé au script via une technique appropriée et permettant de mettre à disposition une fonction ou un contenu similaire à celui proposé par script.

Note : lorsqu'une alternative à un procédé ou une fonctionnalité JavaScript est proposée, le moyen d'y accéder doit être fourni par le site lui-même. Il peut s'agir d'un lien ou d'un bouton permettant d'accéder à une page alternative fonctionnant sans JavaScript ou permettant de remplacer le(s) composant(s) par un composant alternatif fonctionnant sans JavaScript par exemple.

| Alternative courte et concise

Les conditions de restitution d'une alternative textuelle via des technologies d'assistance (par exemple une loupe d'écran) nécessitent qu'elle soit la plus courte possible. Une longueur maximale de 80 caractères est fortement recommandée ; elle limitera le nombre de manipulations nécessaires pour lire l'alternative par les utilisateurs de plages braille ou de loupes d'écran notamment.

| Alternative textuelle (image)

« Nom accessible » restitué par les technologies d'assistance pour les éléments graphiques de type :

- Image (balise `` ou balise ouvrante possédant un attribut WAI-ARIA `role="img"`) ;
- Zone d'image réactive (balise `<area>`) ;
- Bouton de type image (balise `<input>` avec l'attribut `type="image"`) ;
- Image objet (balise `<object type="image/...">`) ;
- Image vectorielle (balise `<svg>`) ;
- Image bitmap (balise `<canvas>`) ;
- Image embarquée (balise `<embed>`).

Dans le cas d'un élément graphique, le « nom accessible » est obtenu selon l'ordre suivant :

- Passage de texte associé via l'attribut WAI-ARIA `aria-labelledby` pour les balises :
 - `;`
 - `<input type="image">;`
 - `<svg>;`
 - `<object type="image/...">;`
 - `<embed type="image/...">;`
 - `<canvas>;`
 - balises possédant un attribut WAI-ARIA `role="img"`.
- Sinon, contenu de l'attribut WAI-ARIA `aria-label` pour les éléments :
 - `;`
 - `<area>;`
 - `<input type="image">;`
 - `<svg>;`
 - `<object type="image/...">;`
 - `<embed type="image/...">;`
 - `<canvas>;`
 - balises ouvrantes possédant un attribut WAI-ARIA `role="img"`.
- Sinon, contenu de l'attribut `alt` pour les balises :
 - `;`
 - `<area>;`
 - `<input type="image">`.
- Sinon, contenu de l'attribut `title` pour les balises :
 - `;`
 - `<input type="image">;`
 - `<object type="image/...">;`
 - `<embed type="image/...">`.

Cet ordre doit être utilisé pour déterminer ce qui constitue l'alternative textuelle.

Néanmoins, en cas de support partiel de l'algorithme de calcul du « nom accessible », c'est la valeur réellement restituée par les technologies d'assistance utilisées dans l'environnement de test (ou « base de référence ») qu'il faudra considérer comme alternative textuelle.

Par exemple :

- En cas de présence conjointe d'un attribut WAI-ARIA `aria-label` et d'un attribut WAI-ARIA `aria-labelledby` sur une balise ``, c'est le passage de texte référencé par l'attribut WAI-ARIA `aria-labelledby` qui doit être considérée comme alternative textuelle si le contenu du passage de texte est réellement restitué par les technologies d'assistance utilisées dans l'environnement de test ;
- En cas de présence conjointe d'un attribut WAI-ARIA `aria-label` et d'un attribut `alt` sur une balise ``, c'est le contenu de l'attribut WAI-ARIA `aria-label` qui doit être considéré comme alternative textuelle si le contenu de l'attribut WAI-ARIA `aria-label` est réellement restitué par les technologies d'assistance utilisées dans l'environnement de test.

Référence : [Accessible name and description calculation](#).

RGAA considère trois types d'alternatives textuelles liées à la nature de l'image :

- Pour une image porteuse d'information, l'alternative textuelle apporte l'information nécessaire à la compréhension du contenu qu'elle véhicule ;
- Pour une image de décoration, aucune alternative textuelle ne doit être restituée ;
- Pour une image [CAPTCHA](#) ou une [image-test](#), l'alternative textuelle décrit seulement la nature et la fonction de l'image. En effet, l'alternative textuelle ne peut apporter l'information véhiculée par l'image sans rendre la fonction associée inopérante.

Note 1 : pour une image [CAPTCHA](#) l'alternative peut être, par exemple : « Code de sécurité anti-spam » ou « code pour vérifier que vous êtes un humain » ou toute autre alternative permettant à l'utilisateur de comprendre la nature et la fonction de l'image.

Note 2 : pour un groupe d'images, par exemple un système de vote constitué de plusieurs images d'étoile, il est fortement conseillé d'utiliser soit la première image du groupe pour donner une alternative cohérente au groupe d'image (voir la [technique WCAG2.1 G196](#)), soit une balise conteneur pourvue d'un rôle WAI-ARIA `img` et d'une alternative textuelle. Dans le premier cas, les autres images du groupe sont considérées comme des images de décoration. Dans le second cas, toutes les images du groupe sont considérées comme des images de décoration.

Note 3 : pour les image-lien, l'alternative doit permettre de comprendre la fonction et la destination du lien ; ce cas est traité dans la thématique liens.

Note 4 : pour les images vectorielles (balise `<svg>`) l'alternative textuelle pourrait se trouver aussi présente dans une balise `<title>` ou dans une balise `<text>` que cette dernière balise soit ou non visible, même si ce n'est pas le rôle dévolu à cet élément en SVG.

Note 5 : l'utilisation de l'attribut `alt` étant la seule technique totalement supportée par les aides techniques il est recommandé de privilégier cette solution lors de la mise en œuvre d'une alternative à une balise ``, `<area>` et `<input type="image">`.

Note 6 : bien que l'attribut `title` soit considéré comme une possibilité d'alternative textuelle à une image, son usage peut poser problème, notamment du fait qu'une image avec un attribut `alt` absent ou vide est considérée comme une image pourvue d'un `role="presentation"` par WAI-ARIA : <https://www.w3.org/TR/html-aam-1.0/#details-id-54>. Il faut s'assurer que les assistances techniques présentes dans l'environnement de test retenu restituent correctement l'alternative proposée par l'attribut `title`.

| Ambigu pour tout le monde

L'intention ne peut être déterminée à partir du lien et de toute l'information de la page web présentée à l'utilisateur en même temps que ce lien (c'est-à-dire qu'un lecteur sans limitation fonctionnelle ne connaîtrait pas la fonction d'un lien avant de l'activer). Exemple : le mot « goyave » dans la phrase suivante utilisé comme lien : « L'une des exportations importantes est la goyave ». Ce lien pourrait conduire à une définition de la goyave, à un graphe présentant une liste des quantités de goyaves exportées ou à une photo de personnes récoltant la goyave. Jusqu'à ce que le lien soit activé, tout utilisateur est dans l'incertitude et une personne handicapée n'est donc pas désavantagée.

| Audiodescription synchronisée (média temporel)

Narration ajoutée (via un fichier son) à une piste sonore pour décrire les détails visuels importants qui ne peuvent être compris à partir de la piste sonore principale seulement. L'audiodescription doit être synchronisée avec le média temporel par un dispositif applicatif lié au lecteur lui-même ou fourni par le développement par exemple avec JavaScript.

Note 1 : l'audiodescription d'une vidéo fournit de l'information à propos des actions, des personnages, des changements de scènes, du texte apparaissant à l'écran et d'autres contenus visuels.

Note 2 : dans une audiodescription standard, la narration est ajoutée durant les pauses qui existent dans le dialogue. (Voir aussi audiodescription étendue.)

Note 3 : lorsque toute l'information de la vidéo est déjà donnée dans la piste audio, aucune audiodescription supplémentaire n'est requise.

2.3.2 B

| Bouton (formulaire)

Élément d'un formulaire qui permet d'effectuer une action prédéfinie. Par exemple, le bouton de soumission d'un formulaire permet l'envoi au serveur des informations collectées pour leur traitement. L'intitulé d'un bouton doit décrire l'action qui résulte de son activation (par exemple : « Lancer votre recherche », « Envoyer votre message »).

En HTML, il y a trois types de boutons de formulaire :

- Balise `<input>` de type `submit`, `reset` ou `button` ;
- Balise `<input>` de type `image` ;
- Balise `<button>`.

Il est également possible de restituer un bouton à l'aide du rôle WAI-ARIA `button`.

L'intitulé du bouton peut être de six types :

- Le contenu du passage de texte associé au bouton via l'attribut WAI-ARIA `aria-labelledby` lorsqu'il est présent ;
- Le contenu de l'attribut `aria-label` lorsqu'il est présent ;
- Le contenu de l'attribut `alt` d'un bouton de type `image` ;
- Le contenu de l'attribut `value` des boutons de type `submit`, `reset` ou `button` ;
- Le contenu de la balise `<button>` ;
- Le contenu de l'attribut `title` lorsqu'il est présent.

Note importante : lorsque plusieurs de ces techniques sont présentes sur un même bouton, le calcul du « nom accessible », c'est-à-dire ce qui sera restitué, obéit à un ordre strict :

- `aria-labelledby` ;
- Sinon `aria-label` ;
- Sinon `alt` pour le cas des `input image` ;
- Sinon `value` pour le cas des `input submit`, `reset` ou `button` ;

- Sinon contenu de la balise `<button>` ;
- Sinon `title`.

Cet ordre doit être utilisé pour l'évaluation de la pertinence du « nom accessible » du bouton. Par exemple, même dans le cas de la présence d'un `title` et d'un passage de texte référencé par `aria-labelledby` sur le même bouton, c'est le passage de texte référencé par `aria-labelledby` qui doit être évalué.

Référence : [Accessible name and description calculation](#).

Par ailleurs, un « nom accessible » sera considéré comme non-pertinent s'il ne reprend pas le texte visible du bouton. Par exemple : `<button aria-label="confirmer la saisie">valider la saisie</button>` sera considéré comme non conforme au [critère 11.9](#).

2.3.3 C

| Cadre

Cadre : élément HTML (balise `<frame>`) permettant d'afficher un contenu dans la page web dans laquelle il est implémenté.

Cadre en ligne : élément HTML (balise `<iframe>`) permettant d'afficher un contenu dans la page web dans laquelle il est implémenté.

| CAPTCHA

Un CAPTCHA est un test utilisé pour distinguer un utilisateur humain d'un ordinateur. Le test utilise souvent des images contenant du texte déformé, mélangé avec d'autres formes ou utilisant des jeux de couleur altérées, que l'utilisateur est invité à retaper. D'autres formes de CAPTCHA peuvent être basées sur des questions logiques ou des extraits sonores.

| Champ de saisie de formulaire

Objet d'un formulaire permettant à l'utilisateur :

- De saisir des données textuelles ou préformatées :
 - `input type="text" ;`
 - `input type="password" ;`
 - `input type="search" ;`
 - `input type="email" ;`
 - `input type="number" ;`
 - `input type="tel" ;`
 - `input type="url" ;`
 - `textarea`.
- De sélectionner des valeurs prédéfinies :
 - `input type="checkbox" ;`
 - `input type="radio" ;`
 - `input type="date" ;`
 - `input type="range" ;`

- `input type="color";`
- `input type="time";`
- `input type="month";`
- `input type="week";`
- `input type="datetime-local";`
- `select;`
- `datalist;`
- `optgroup;`
- `option.`
- **De télécharger des fichiers :**
 - `input type="file".`
- **Ou d'afficher des résultats :**
 - `output;`
 - `progress;`
 - `meter.`
- **Les balises possédant un rôle WAI-ARIA permettant de restituer un champ de formulaire sont également couvertes par cette définition :**
 - `progressbar;`
 - `slider;`
 - `spinbutton;`
 - `textbox;`
 - `listbox;`
 - `searchbox;`
 - `combobox;`
 - `option;`
 - `checkbox;`
 - `radio;`
 - `switch.`
- **Les objets de formulaires et rôle WAI-ARIA suivants ne sont pas considérés comme des champs de formulaires :**
 - `input type="submit";`
 - `input type="reset";`
 - `input type="hidden";`
 - `input type="image";`
 - `input type="button";`
 - `button;`
 - **attribut WAI-ARIA `role="button".`**

Champs de même nature

Dans un formulaire, ensemble des champs pouvant être regroupés par la nature des informations attendues. Le regroupement vise à identifier les champs devant être traités comme un ensemble.

Quelques exemples :

- Trois champs successifs pour saisir une date (jour/mois/année) ;
- Champs successifs pour un numéro de téléphone ;
- Un bloc destiné à saisir l'identité et l'adresse de l'utilisateur, lorsque le formulaire contient plusieurs blocs de contact ;
- Un ensemble de boutons radio ou de cases à cocher qui se rapportent à une question.

Ces champs doivent être regroupés lorsque les intitulés de label ne sont pas suffisants pour informer l'utilisateur que les champs font partie d'un regroupement.

Changement brusque de luminosité ou effet de flash

Alternance de luminance relative qui peut causer des crises chez certaines personnes si leur taille est suffisamment importante dans une gamme de fréquences spécifiques.

Changement de contexte

Changements majeurs dans le contenu d'une page web qui, s'ils sont faits sans que l'utilisateur n'en soit conscient, peuvent désorienter l'utilisateur qui ne peut voir l'ensemble de la page en même temps. Les changements de contexte comprennent les changements :

- d'agent utilisateur ;
- d'espace de restitution ;
- de focus ;
- de contenu qui modifie la signification de la page web.

Note : Un changement de contenu n'est pas toujours un changement de contexte. Un changement dans le contenu comme le déploiement d'une arborescence, un menu dynamique ou un déplacement de tabulation ne change pas nécessairement le contexte à moins qu'il ne change aussi l'un des éléments énumérés ci-dessus (le focus, par exemple).

Par exemple, l'ouverture d'une nouvelle fenêtre, le déplacement du focus sur un composant différent, le déplacement vers une nouvelle page (y compris tout ce qui, pour l'utilisateur, aurait l'air d'un déplacement vers une autre page) ou la réorganisation significative du contenu d'une page sont autant d'illustrations d'un changement de contexte.

Changement de langue

L'indication des changements de langue est nécessaire pour indiquer aux technologies d'assistance de modifier la restitution vocale d'un élément. Les changements de langue concernent tous les contenus, y compris les valeurs de certains attributs comme `title`.

Note : il n'est pas possible d'indiquer des changements de langue dans une valeur d'attribut elle-même, dans ce cas le changement de langue est indiqué sur l'élément qui contient l'attribut. Par exemple un lien affecté d'un `title` en anglais devra comporter un attribut `lang="en"`. Lorsque l'attribut contient plusieurs passages de texte dans des langues différentes, le critère est non applicable.

Code de langue

Code de 2 caractères (ISO 639-1) ou 3 caractères (ISO 639-2 et suivants) permettant d'indiquer la langue d'un document ou d'un passage de texte. L'indication du code de langue est constituée de deux parties séparées par un tiret sur le modèle `lang="[code]-[option]"`.

- [code] représente un code de langue valide sur 2 ou 3 caractères ;
- [option] est une indication laissée à l'appréciation de l'auteur.

Lorsqu'un code de pays est utilisé comme option, il peut servir à indiquer une régionalisation de la langue, l'indication "en-us" indique la langue américaine, par exemple. L'indication du code de langue ne concerne que la partie [code] avant le tiret.

Compatible avec les technologies d'assistance

Un contenu ou une fonctionnalité doit être compatible avec les technologies d'assistance des utilisateurs ainsi qu'avec les fonctions d'accessibilité des navigateurs et des autres agents utilisateurs via une API d'accessibilité.

Cela concerne, à la fois, la technologie, ses fonctionnalités et ses usages :

- La façon dont la technologie Web est utilisée doit être compatible avec les technologies d'assistance des utilisateurs. Cela signifie que la façon dont la technologie est utilisée a été testée dans une perspective d'interopérabilité avec des utilisateurs des technologies d'assistance dans la ou les langues du contenu ;
- La technologie fonctionne de façon native dans des agents utilisateurs largement distribués qui sont, eux-mêmes, compatibles avec l'accessibilité (comme HTML et CSS) ou avec un module d'extension largement distribué qui est, lui-même, compatible avec l'accessibilité.

La vérification de la compatibilité avec les technologies d'assistance nécessite de réaliser un certain nombre de tests spécifiques à la technologie utilisée, par exemple :

- Vérifier le nom, le rôle, le paramétrage et les changements d'états des composants d'interface ;
- Vérifier que la restitution d'un composant d'interface est correcte pour la ou les technologies d'assistance utilisées.

Composant d'interface

Un composant d'interface est un élément avec lequel l'utilisateur peut interagir, par exemple un bouton, un lien, une zone de saisie. Certains composants peuvent être plus complexes comme un menu, une fenêtre de dialogue, un système d'onglets. Enfin, un composant d'interface peut être basé sur des éléments natifs de HTML ou développés de toutes pièces en JavaScript et des attributs WAI-ARIA. En particulier pour les éléments ayant des attributs WAI-ARIA correspondant à un [motif de conception](#) il est recommandé de considérer le document [WAI-ARIA 1.1 Authoring Practices](#) lors de leur implémentation.

Compréhensible (ordre de lecture)

Un contenu compréhensible est lisible (l'ordre des éléments est logique) et cohérent (l'enchaînement de la lecture est cohérent).

Contenu alternatif

Contenu venant se substituer à un autre apportant la même information mais pouvant être présenté de façon différente. Ce contenu peut être de forme textuelle mais également être lui-même structuré à l'aide de balises. Un contenu alternatif devra respecter l'ensemble des critères du RGAA qui lui sont applicables pour être considéré comme une alternative accessible à l'élément qu'il remplace. Exemple : un tableau de données peut être le contenu alternatif d'une image bitmap (balise `<canvas>`) affichant un graphique statistique.

Contenu caché

Les technologies d'assistance (notamment les lecteurs d'écran) ne restituent pas le contenu masqué via les propriétés :

- `display` avec la valeur `none` (`display: none`) ;
- `visibility` avec la valeur `hidden` (`visibility: hidden`) ;
- `font-size` avec la valeur `0` (`font-size:0`) ;
- Attribut HTML5 `hidden` ;
- Attribut WAI-ARIA `aria-hidden="true"`.

Tous les contenus utilisant une ou plusieurs de ces propriétés et attributs sont applicables pour le [critère 10.8](#).

Contenu visible

Pour le [test 12.2.1](#) : « Contenu présent » signifie que le contenu visible reste présent lorsque CSS est désactivé. Par exemple, une image porteuse d'information en propriété de fond CSS invalide ce test car l'information n'est plus « présente » lorsque CSS est désactivé. En revanche, une image porteuse d'information en propriété de fond CSS mais accompagnée d'un texte caché valide ce test car l'information est bien « présente » lorsque CSS est désactivé.

Note : la pratique qui consiste à gérer des images en propriété de fond d'éléments via CSS est formellement déconseillée, même si elle est accompagnée d'un texte caché.

Contexte du lien

Le contexte du lien représente les informations supplémentaires (on parle d'informations de contexte) qui peuvent être mises en relation par un programme informatique avec l'[intitulé du lien](#). Les informations de contexte qui permettent de compléter l'[intitulé du lien](#) sont les suivantes :

- Le contenu de la phrase dans laquelle le lien texte est présent ;
- Le contenu du paragraphe (balise `<p>`) dans lequel le lien texte est présent ;
- Le contenu de l'item de liste (balise ``) ou le contenu d'un item de liste parent (balise ``) dans lequel le lien texte est présent ;
- Le contenu du titre (balise `<h>`) précédent le lien texte ;
- Le contenu de la ou les cellule(s) d'en-tête de tableau (balise(s) `<th>`) associée(s) à la cellule de donnée (balise `<td>`) dans laquelle le lien texte est présent ;
- Le contenu de la cellule de donnée (balise `<td>`) dans laquelle le lien texte est présent.

Note 1 : l'un des 6 contextes de lien combiné à l'[intitulé du lien](#) doit permettre de comprendre la fonction et la destination du lien.

Note 2 : RGAA considère qu'une adresse e-mail de type `xxx@xxx.yyy` est un texte de lien suffisant pour comprendre la fonction du lien et ne requiert pas de signaler plus explicitement l'action.

Contraste

Opposition marquée entre la luminosité d'une couleur de premier plan et d'une couleur d'arrière-plan. Le rapport de contraste est basé sur la différence de luminance relative entre l'arrière-plan et le premier plan selon la règle : $(L1 + 0,05) / (L2 + 0,05)$ où L1 est la luminance relative la plus claire et L2 la luminance relative la plus sombre. La luminosité est calculée selon la formule suivante : $L = 0,2126 * R + 0,7152 * G + 0,0722 * B$. Où R, G et B sont définis par :

- Si $R_{sRGB} \leq 0,03928$ alors $R = R_{sRGB} / 12,92$ sinon $R = ((R_{sRGB} + 0,055) / 1,055) ^ 2,4$;
- Si $G_{sRGB} \leq 0,03928$ alors $G = G_{sRGB} / 12,92$ sinon $G = ((G_{sRGB} + 0,055) / 1,055) ^ 2,4$;
- Si $B_{sRGB} \leq 0,03928$ alors $B = B_{sRGB} / 12,92$ sinon $B = ((B_{sRGB} + 0,055) / 1,055) ^ 2,4$.

et R_{sRGB} , G_{sRGB} et B_{sRGB} sont définis par :

- $R_{sRGB} = R_{8bit} / 255$;
- $G_{sRGB} = G_{8bit} / 255$;
- $B_{sRGB} = B_{8bit} / 255$.

Le caractère « ^ » est l'opérateur de puissance.

Note : la mesure de contraste concerne le texte, le texte en image, le texte et le texte en image dans les animations, le texte de sous-titrage et le texte incrusté dans les vidéos. Pour le texte et le texte en image dans les animations, le texte de sous-titrage et le texte incrusté dans les vidéos, la taille de la police doit être mesurée par rapport à la taille d'affichage par défaut (telle qu'affichée). Les textes présents dans les éléments d'une image ou d'une vidéo (par exemple un écriteau, une affiche...) ne sont pas concernés.

Source (en anglais) : [Procédure de calcul de contraste des WCAG](#).

Contrôle (contenu en mouvement ou clignotant)

Possibilité pour l'utilisateur de contrôler l'affichage ou la lecture d'un contenu en mouvement ou clignotant par le clavier et la souris, au moins.

Tous les contenus en mouvement, à l'exception des médias temporels pris en charge par la thématique multimédia, sont concernés : les images animées (par exemple un gif animé), les contenus en mouvement proposés via une balise `<object>`, du code JavaScript ou des effets CSS par exemple.

Note 1 : lorsque c'est approprié, la méthode de contrôle devrait être disponible comme premier élément de la page.

Note 2 : la méthode de contrôle du contenu en mouvement ou clignotant doit permettre à l'utilisateur d'interagir avec le reste de la page. En conséquence, l'arrêt ou la mise en pause via un événement déclenché uniquement sur la prise de focus ne permet pas de valider le critère.

Note 3 : Dans certains cas, le mouvement fait partie intégrante du composant et il n'est pas possible d'en donner le contrôle à l'utilisateur, par exemple une barre de progression dont la fonction est d'indiquer par un mouvement la progression d'un événement comme un téléchargement. Dans ce cas le critère est Non Applicable.

Contrôle de la consultation (d'un média temporel)

Possibilité pour l'utilisateur de contrôler la consultation d'un média temporel par le clavier et tout dispositif de pointage, au moins. Les points suivants doivent être respectés :

- Liste des fonctionnalités obligatoires de contrôle de la consultation :
 - L'objet multimédia doit toujours avoir les fonctionnalités suivantes, au minimum : lecture, pause ou stop ;
 - Si l'objet multimédia a du son, il doit avoir une fonctionnalité d'activation / désactivation du son ;
 - Si l'objet multimédia a des sous-titres, il doit avoir une fonctionnalité de contrôle de l'apparition / disparition des sous-titres ;
 - Si l'objet multimédia a une audiodescription, il doit avoir une fonctionnalité de contrôle de l'apparition / disparition de l'audiodescription.
- Chaque fonctionnalité doit être accessible par le clavier, via la touche de tabulation, et par tout dispositif de pointage au moins.
- Chaque fonctionnalité doit être activable par le clavier et par tout dispositif de pointage, au moins.

Note : s'il n'y a pas de son à un média temporel, il n'est pas utile de mettre une fonctionnalité d'activation / désactivation du son. Si cette fonctionnalité est cependant présente et qu'elle nécessite une alternative textuelle pour être comprise par certains utilisateurs, il faut alors lui en donner une puisque l'utilisateur est susceptible d'y accéder et de vouloir l'activer.

Contrôle de saisie (formulaire)

Ensemble des processus qui permettent de prévenir l'utilisateur des champs obligatoires, des indications de type ou de format attendus et des erreurs de saisie dans un formulaire. Ces contrôles de saisie peuvent être implémentés par l'auteur des contenus ou s'appuyer sur des attributs (comme `required` ou `pattern`), des attributs WAI-ARIA (comme `aria-required`) ou des types de champ qui produisent de manière automatique des indications de saisie ou d'erreurs (comme les types `url`, `email`, `date`, `time` par exemple).

Contrôle (son déclenché automatiquement)

Possibilité pour l'utilisateur d'arrêter ou de relancer un son déclenché automatiquement.

Note : la méthode de contrôle du son devrait être disponible comme premier élément de la page.

Correctement restitué (par les technologies d'assistance)

Lorsqu'un critère, un test ou une condition de test demande de vérifier la restitution d'un dispositif, il faut s'assurer que ladite restitution est compatible avec l'accessibilité.

Le test consiste à vérifier que la restitution est pertinente pour au moins une des combinaisons de l'environnement de test (ou « base de référence ») utilisé pour déclarer qu'un élément, un dispositif ou une alternative est « compatible avec l'accessibilité ».

Par exemple : le [test 1.3.8](#) demande de vérifier que l'alternative d'une image bitmap (balise `<canvas>`) porteuse d'information est correctement restituée.

On procède alors à un test avec les outils de l'environnement de test défini pour le site.

Si on constate que l'alternative est correctement restituée, le test est validé.

Couleur d'arrière-plan contiguë et couleur contiguë

Couleur d'arrière-plan contiguë : couleur d'arrière-plan directement en contact avec le bord extérieur du composant d'interface ou de l'[élément graphique](#).

Exemples :

- Pour un bouton blanc avec une bordure bleue sur un fond blanc, le fond blanc à l'extérieur de la bordure bleue correspond à la couleur d'arrière-plan contiguë ;
- Pour un bouton rouge sur fond blanc, le fond blanc à l'extérieur du rouge correspond à la couleur d'arrière-plan contiguë ;
- Pour un bouton blanc avec une bordure verte qui devient noire à la prise de focus et au survol, le fond blanc à l'extérieur de la bordure verte de l'état par défaut et de la bordure noire de l'état au survol et au focus correspond à la couleur d'arrière-plan contiguë.

Couleur contiguë : couleur directement en contact avec une autre couleur. Exemple dans un panneau de « sens interdit » le rouge du panneau est la couleur contiguë au trait blanc au centre du panneau.

Note 1 : dans le cas de la présence d'un dégradé, c'est la couleur contiguë la moins contrastée du dégradé qui sera à considérer comme la couleur contiguë ou couleur d'arrière-plan contiguë.

Note 2 : dans le cas de la présence de plusieurs couleurs, c'est l'ensemble des couleurs qui seront à considérer comme couleur contiguë ou couleur d'arrière-plan contiguë.

2.3.4 D

Description détaillée (image)

Contenu associé à une image en complément de son alternative textuelle afin de décrire en totalité l'information véhiculée par l'image. La description détaillée peut être associée à l'image via :

- Un attribut `longdesc` qui contient l'adresse d'une page ou d'un emplacement dans la page contenant la description détaillée ;
- Une référence à une description détaillée adjacente à l'image dans l'alternative textuelle ;
- Un [lien ou un bouton adjacent](#) qui permet d'accéder à la description détaillée dans la page ou dans une autre page ;
- Un ou plusieurs passages de texte identifiés par un `id` et liés par un attribut WAI-ARIA `aria-describedby` sur le modèle `aria-describedby="ID1 ID2 ID3..."`.

Note 1 : Si le support de l'attribut `aria-describedby` fait défaut, il est possible d'utiliser un ou plusieurs passages de texte identifiés par un `id` et liés par un attribut WAI-ARIA `aria-labelledby` à la suite de l'alternative textuelle.

Note 2 : Pour assurer une compatibilité maximum avec les agents utilisateurs, notamment Internet Explorer 11, il est recommandé d'implémenter un `tabindex="-1"` sur les balises qui contiennent un passage de texte et qui ne sont pas des éléments interactifs (bouton, liens, éléments de formulaires, etc.).

2.3.5 E

Élément graphique

Élément faisant appel à une représentation visuelle telle que des images, des pictogrammes ou des graphiques.

Cet élément peut être composé d'une ou plusieurs parties dont la visibilité est nécessaire à sa compréhension (par exemple chaque point sur chaque ligne d'un graphique de statistiques).

Ensemble de pages

Pages web liées les unes aux autres par des liens et constituant un ensemble cohérent à l'intérieur d'un site web. Par exemple, les pages d'une rubrique spécifique, les pages d'un blog, les pages d'administration d'un compte client sont autant d'ensembles de page.

Note : la page d'accueil d'un site web peut constituer, à elle seule, un « ensemble de pages » du fait de son unicité.

En-tête de colonne ou de ligne

Contenu d'une cellule dans un tableau de données (la première cellule d'une colonne ou d'une ligne, généralement) qui sert d'intitulé pour la totalité ou une partie des cellules de la colonne ou de la ligne. Une colonne ou une ligne peut contenir plusieurs en-têtes (en-tête intermédiaire). Lorsque les en-têtes s'appliquent à l'ensemble d'une ligne ou d'une colonne, ils peuvent être structurés avec une balise `<th>` ou une balise pourvue d'un attribut WAI-ARIA `role="rowheader"` ou `role="columnheader"`. Dans le cas contraire, seule une balise `<th>` peut être utilisée.

Note : seule la balise `<th>` étant totalement supportée par l'ensemble des technologies d'assistance, il est fortement recommandé de privilégier cette solution lors de la mise en œuvre afin d'éviter de nombreuses vérifications dans les différentes combinaisons prévues dans l'environnement de test (ou « base de référence »).

Environnement maîtrisé

Tout environnement dans lequel l'accès à l'information, les technologies, les conditions d'utilisation et le profil des utilisateurs peuvent être connus et maîtrisés. Les principaux éléments dont la maîtrise est essentielle sont :

- Le type et la version des navigateurs ;
- Les technologies supportées, leur version et leur activation (JavaScript, WAI-ARIA, Flash, Silverlight...);
- Les technologies d'assistance et tout dispositif utilisé de manière spécifique par les utilisateurs handicapés ;
- Les systèmes d'exploitation et les APIs d'accessibilité supportées ;
- La formation des utilisateurs de technologies d'assistance à l'utilisation de tout dispositif particulier (interface, application en ligne...).

Les auteurs et les administrateurs doivent garantir la compatibilité des technologies utilisées et de leurs usages par les utilisateurs et leurs technologies (y compris les technologies d'assistance). Les

services d'information ou les sites Web, quel que soit leur statut, qui offrent un accès public ne peuvent pas être considérés comme des environnements maîtrisés.

Étiquette de champ de formulaire

Texte à proximité du champ de formulaire permettant d'en connaître la nature, le type ou le format des informations attendues. L'étiquette peut être associée au champ de formulaire de plusieurs manières :

- Par l'utilisation d'une balise `<label>` ;
- Par l'utilisation de l'attribut WAI-ARIA `aria-label` ;
- Par l'utilisation d'une liaison entre le texte et le champ par l'attribut WAI-ARIA `aria-labelledby` ;
- Par l'utilisation de l'attribut `title`.

Note importante : lorsque plusieurs de ces techniques sont présentes sur un même champ, le calcul du « nom accessible », c'est-à-dire ce qui sera restitué, obéit à un ordre strict :

- `aria-labelledby` ;
- Sinon `aria-label` ;
- Sinon `<label>` ;
- Sinon `title`.

Cet ordre doit être utilisé pour l'évaluation de la pertinence de l'étiquette ([critère 11.2](#)). Par exemple, même dans le cas de la présence d'un `<label>`, c'est le passage de texte référencé par `aria-labelledby` qui doit être pris en compte.

Référence : [Accessible name and description calculation](#).

Note importante au sujet de l'utilisation de `placeholder` : lorsque l'attribut `placeholder` est présent, il est susceptible d'être restitué à la place de l'attribut `title`. Par conséquent, lorsque ces deux attributs `title` et `placeholder` sont présents, ils doivent être identiques.

Note au sujet des étiquettes liées par `aria-labelledby` : Il s'agit d'un ou de plusieurs passages de texte identifiés par des `id` et liés par `aria-labelledby` sur le modèle suivant : `aria-labelledby="ID1 ID2 ID3..."`. Pour assurer une compatibilité maximum avec les agents utilisateurs, notamment Internet Explorer 11, il est recommandé d'implémenter un `tabindex="-1"` sur les passages de textes qui ne sont pas des éléments interactifs (bouton, liens, éléments de formulaires, etc.).

Note : l'attribut `aria-label` ne peut pas être utilisé pour indiquer le caractère obligatoire d'un champ.

Étiquettes cohérentes

Les étiquettes de champs de formulaire présentes dans une même page ou dans un ensemble de pages et réclamant la saisie d'une même information doivent être formulées sans ambiguïté pour que l'utilisateur sache que l'information qu'il doit communiquer est la même.

2.3.6 F

Feuille de style

Le langage CSS destiné à la mise en forme des éléments du contenu (exemples : couleur du fond de la page, taille / police / couleur des caractères, positionnement de l'information dans la page web...). Les styles CSS peuvent être externes (fichier CSS), embarqués (déclarés dans l'en-tête de la page) ou en ligne (déclarés via l'attribut `style` d'une balise).

| Formulaire

Balise `<form>` ou balise possédant un attribut WAI-ARIA `role="form"`.

2.3.7 G

| Gestes complexes et gestes simples

Un geste simple implique un contact en un point unique de l'écran. Il peut s'agir d'une pression ou d'un clic simple, d'une double-pression ou d'un double-clic, d'une pression prolongée.

Un geste complexe peut être à la fois un geste impliquant plusieurs points de contact sur l'écran (exemple : un geste avec deux doigts sur l'écran pour zoomer ou dézoomer une carte) et un geste basé sur le suivi d'une trajectoire sur l'écran (exemple : fonction JavaScript permettant de détecter le déplacement d'un doigt vers la gauche ou droite sur une surface tactile pour déclencher le passage à l'item précédent / suivant d'un carrousel).

2.3.8 I

| Image de décoration

Image n'ayant aucune fonction et ne véhiculant aucune information particulière par rapport au contenu auquel elle est associée.

Exemples :

- Une image précédant chaque item d'une liste ;
- Une image servant à caler la mise en page ;
- Une image de coin arrondie pour habiller un bloc d'information ;
- Une image d'illustration n'apportant aucune information nécessaire à la compréhension du texte auquel elle est associée.

Note : les balises possédant un attribut WAI-ARIA `role="img"` ne peuvent faire office d'image de décoration qu'à la condition qu'elles possèdent un attribut WAI-ARIA `aria-hidden="true"`.

| Image objet

Image incorporée ou générée par une balise `<object>`.

| Image porteuse d'information

Image qui véhicule une information nécessaire à la compréhension du contenu auquel elle est associée.

Note 1 : lorsque l'image est le seul contenu d'un lien, son alternative est l'intitulé du lien. Dans ce cas, l'alternative de l'image devrait être évaluée avec la thématique « Liens ».

Note 2 : lorsqu'un bouton de formulaire, inséré avec l'élément `<button>`, ne contient qu'une image (balise ``, `<object>`, `<embed>`, `<canvas>` ou `<svg>`), l'alternative de l'image est l'intitulé du bouton. Deux cas peuvent se présenter :

- Le bouton est contrôlé par son type, par exemple, le type `submit` ou `reset`, et fait partie d'un formulaire. Dans ce cas, le bouton image doit être évalué avec la thématique « Formulaires » ;
- Le bouton est contrôlé par un dispositif JavaScript. Dans ce cas, le bouton image doit être évalué avec la thématique « Scripts ».

| Image réactive

- Image réactive côté client (attribut `usemap`) : image divisée en zones cliquables ou neutres (attribut `nohref`) ;
- Image réactive côté serveur (attribut `ismap`) : image pour laquelle le navigateur transmet au serveur les coordonnées du pointeur, chaque jeu de coordonnées correspondant à une ressource (page web). L'image réactive côté serveur est extrêmement rare.

Note : en HTML5, l'attribut `ismap` est obsolète et non conforme pour les boutons de type `image` (`input type="image"`).

| Image-test

Image servant dans un test, captcha ou une image servant de test dans un quiz ou un jeu.

Exemple : une série d'images présente un détail issu de tableaux célèbres ; il faut reconnaître le titre et le peintre de chaque tableau. Dans cette situation, il n'est pas possible de donner une alternative pertinente (par exemple le nom du tableau et/ou du peintre) sans rendre le test inutilisable.

L'alternative doit alors se contenter de donner la possibilité d'identifier l'image, par exemple "image 1 du test".

| Image texte

Image affichant du texte.

Note : il n'est pas recommandé d'utiliser des images-textes. Lorsqu'il est possible de reproduire les mêmes effets en CSS, le [critère 1.8](#) impose que le texte soit reproduit en texte CSS, ou qu'un mécanisme de remplacement permette à l'utilisateur de remplacer ces images par du [texte stylé](#) en CSS.

| Image texte objet

Image générée par la balise `<object>` et affichant du texte.

| Image véhiculant une information (donnée par la couleur)

Image dont tout ou partie du contenu transmet visuellement une information par l'intermédiaire d'une couleur uniquement.

| Indication de champ obligatoire

Indication textuelle ou graphique (icône) permettant à l'utilisateur de savoir que la saisie d'un champ est obligatoire préalablement à la saisie.

Note : Dans le cas où cette indication n'est pas réalisée de manière textuelle explicite (icône, "***", "!", etc.), l'explication de la signification de cette indication doit se situer, visuellement et dans l'ordre du code source, avant la première utilisation de l'indication.

Indication donnée par la forme, la taille ou la position

Il peut s'agir, par exemple :

- De la présence d'un marqueur visuel, pour indiquer la page active dans un menu de navigation (indication donnée par la position) ;
- D'une mise en avant-plan pour indiquer un onglet actif (indication donnée par la forme) ;
- D'une modification de la taille de police dans un nuage de tags (indication donnée par la taille).

Ou de tout autre effet graphique similaire.

Indication du type de données et/ou de format

Indication textuelle permettant à l'utilisateur de savoir quel est le type de donnée et/ou le format de saisie requis par un champ obligatoire, préalablement à son renseignement.

Exemples :

- Courriel (format : vous@domaine.com) ;
- Code postal (format : 00000) ;
- Date (format : JJ/MM/AAAA).

Information (donnée par la couleur)

Information transmise visuellement par l'intermédiaire d'une couleur. L'indication que les champs en rouge sont obligatoires dans un formulaire, l'utilisation d'un fond bleu pour indiquer la page en cours de consultation dans un menu avec le fond vert, le changement de couleur d'un nom d'article pour indiquer son indisponibilité dans une liste d'articles sont autant d'exemples d'indication donnée par la couleur.

Lorsqu'une information donnée par la couleur est accompagnée d'une autre méthode à destination des utilisateurs qui ne voient pas ou perçoivent mal les couleurs ou leurs associations, le critère sera considéré comme non applicable.

Les moyens de transmettre une information autrement que par la couleur peuvent être :

- Une indication textuelle visible ;
- Un moyen faisant intervenir du graphisme (pictogramme, image de fond, forme, style de bordure différent, etc.) et par le biais d'un complément au niveau du code (`aria-label`, `title`, `texte masqué`, `aria-current`, etc.) ;
- Un autre style typographique (gras, italique, taille de texte, autre police, etc) et par le biais d'un complément au niveau du code (`aria-label`, `title`, `texte masqué`, `aria-current`, etc.).

Intitulé (ou nom accessible) de lien

« Nom accessible » restitué par les technologies d'assistance.

Dans le cas d'un lien HTML, le « nom accessible » est obtenu selon l'ordre suivant :

- passage de texte associé par l'attribut WAI-ARIA `aria-labelledby` ;
- sinon, contenu de l'attribut WAI-ARIA `aria-label` ;
- sinon, contenu du lien ;
- sinon, contenu de l'attribut `title`.

Cet ordre doit être utilisé pour déterminer ce qui constitue l'intitulé du lien. Par exemple :

- en cas de présence conjointe d'un attribut WAI-ARIA `aria-label` et d'un attribut WAI-ARIA `aria-labelledby`, c'est le passage de texte référencé par l'attribut WAI-ARIA `aria-labelledby` qui doit être considéré comme l'intitulé ;
- en cas de présence conjointe d'un attribut WAI-ARIA `aria-label` et d'un contenu dans le lien, c'est le contenu de l'attribut WAI-ARIA `aria-label` qui doit être considéré comme l'intitulé.

Référence : [Accessible name and description calculation](#).

Dans le cas où le « nom accessible » est obtenu à partir du contenu du lien, celui-ci sera variable en fonction des cas suivants :

Lien texte :

En HTML, le « nom accessible » correspond au texte constitué à partir :

- du texte contenu dans le lien ;
- du texte contenu dans les éléments enfants du lien.

Lien image :

En HTML, le « nom accessible » correspond au texte constitué à partir de l'alternative textuelle d'une ou plusieurs images dans le lien du type :

- Image (élément `` ou balise ouvrante ayant l'attribut WAI-ARIA `role="img"`) ;
- Image objet (élément `<object>`) ;
- Image bitmap (élément `<canvas>`) ;
- Image vectorielle (élément `<svg>`).

Lien composite :

En HTML, le « nom accessible » correspond au texte constitué à partir de l'ensemble :

- du texte contenu dans le lien ;
- du texte contenu dans les éléments enfant du lien ;
- du contenu de l'alternative textuelle de la ou des images comprises dans le lien.

Dans le cas d'un lien SVG (version 1.1), le « nom accessible » est obtenu comme suit :

- Passage de texte associé par l'attribut WAI-ARIA `aria-labelledby` ;
- Sinon, contenu de l'attribut WAI-ARIA `aria-label` ;
- Sinon, contenu de l'élément `<title>` enfant direct du lien ;
- Sinon, contenu de l'attribut `xlink:title` ;

- Sinon, contenu texte d'un ou plusieurs éléments `<text>`.

Il faut cependant être vigilant car cet algorithme de calcul n'est pas encore pris en compte et effectif au sein des différents lecteurs d'écran. À ce jour, le support est disponible avec VoiceOver, mais incomplet ou lacunaire avec JAWS et NVDA. Si bien que le plus petit dénominateur commun sur lequel il est possible de se reposer pour fournir un intitulé au lien est l'élément `<text>`.

Note 1 : un intitulé de lien sera considéré comme non-explicite dans le cas où le « nom accessible » ne reprend pas l'[intitulé visible](#) du lien.

Note 2 : en raison de la configuration possible des aides techniques permettant de forcer la restitution du « nom accessible » issu du contenu de l'attribut `title` au détriment du « nom accessible » issu du contenu du lien. Un intitulé de lien sera considéré comme non-explicite dans le cas où le lien possède un attribut `title` dont la valeur ne reprendrait pas au moins le « nom accessible » issu du contenu du lien.

Note 3 : dans le cas de la présence de plusieurs liens ayant une destination différente dont le « nom accessible » est identique. L'intitulé de lien seul sera considéré comme non-explicite si le contexte de lien ne permet pas de les différencier.

Note 4 : lorsqu'un lien ne comporte aucun contenu, il sera non conforme au regard du [critère 10.2](#) et du [critère 6.2](#).

Note 5 : bien que le calcul du nom accessible d'un lien tienne compte de contenus texte générés en CSS via les pseudo-éléments `::before` et `::after`, cette pratique ne doit pas être utilisée, car elle constitue une non-conformité au critère 1.3.1 des WCAG 2.1 (cf. [F87](#)).

| Intitulé visible

Texte affiché faisant office d'intitulé visible à l'écran au sein d'un bouton ou d'un lien.

Texte affiché faisant office d'[étiquette](#) pour un champ formulaire.

Ce texte peut être constitué de texte ou d'une image contenant du texte.

| Items de même nature d'une liste de choix

Dans une liste déroulante (balise `<select>`), ensemble d'items (balises `<option>`) pouvant être regroupés par leur nature. Le regroupement vise à identifier les items devant être traités comme un ensemble (par exemple, une liste de départements regroupés par régions).

2.3.9 L

| Landmarks

WAI-ARIA propose des rôles permettant d'indiquer les zones principales (régions) du document. Ces rôles sont très profitables aux utilisateurs de lecteurs d'écran notamment, mais également aux utilisateurs de la navigation au clavier qui peuvent ainsi bénéficier de fonctionnalités de navigation rapide dans la [structure du document](#).

Les rôles doivent être définis dans le document en fonction de la nature de la zone :

- La zone d'[en-tête](#) doit avoir un attribut WAI-ARIA `role="banner"` ;
- Le [menu de navigation](#) principal doit avoir un attribut WAI-ARIA `role="navigation"` ;
- La zone de [contenu principal](#) doit avoir un attribut WAI-ARIA `role="main"` ;
- La zone de [pied de page](#) doit avoir un attribut WAI-ARIA `role="contentinfo"` ;
- La zone de [moteur de recherche](#) sur le site doit avoir un attribut WAI-ARIA `role="search"`.

Note 1 : Si la plupart des lecteurs d'écran mettent à disposition ces fonctionnalités, les navigateurs n'ont pas encore proposé de fonctionnalité de navigation dédiée pour les utilisateurs qui ne peuvent pas utiliser la souris. La mise en place des liens d'évitement reste donc à privilégier par rapport aux landmarks.

Note 2 : Les rôles WAI-ARIA `banner`, `main` et `contentinfo` doivent être uniques dans la page. Le rôle WAI-ARIA `navigation` est réservé aux zones de navigations principales et secondaires. Lorsqu'il y a plusieurs rôles WAI-ARIA `navigation`, il peut être utile de les différencier en précisant un nom à chacune des zones via l'attribut WAI-ARIA `aria-label` ou `aria-labelledby`.

| Langue par défaut

Indication de la langue de traitement principale du document qui peut être présente sur l'élément racine `html` ou sur chaque élément de la page concerné via les attributs `lang` et/ou `xml:lang` selon le schéma suivant :

- Pour HTML jusqu'à la version 4.01 : attribut `lang` obligatoire, attribut `xml:lang` non supporté ;
- Pour XHTML 1.0 servi en `"text/html"` : attribut `lang` et `xml:lang` obligatoires ;
- Pour XHTML 1.0 servi en `"application/xhtml+xml"` : attribut `xml:lang` obligatoire, attribut `lang` recommandé ;
- Pour XHTML 1.1 : attribut `xml:lang` obligatoire, attribut `lang` non supporté ;
- Pour HTML5 : attribut `lang` obligatoire.

| Légende

HTML propose un dispositif permettant de titrer les groupes de champs de même nature par l'intermédiaire des éléments `<fieldset>` et `<legend>`.

Il est également possible de créer des regroupements avec le rôle WAI-ARIA `group` et un passage de texte, faisant office de légende, lié par l'attribut WAI-ARIA `aria-labelledby` ou fourni par un attribut WAI-ARIA `aria-label`.

Note 1 : Les regroupements de champs peuvent utiliser d'autres méthodes qui associent l'information du regroupement directement dans l'étiquette du champ. Par exemple, par l'intermédiaire d'un attribut `title`, d'un attribut WAI-ARIA `aria-label`, d'une liaison `aria-labelledby` faisant office d'étiquette ou encore par l'attribut WAI-ARIA `aria-describedby` associant un texte complémentaire. Dans ce cas, le regroupement de champs devient inutile puisque les labels sont suffisamment pertinents.

Note 2 : Lorsque le formulaire est constitué d'un seul bloc d'informations de même nature (l'identité et l'adresse de l'utilisateur, par exemple) ou d'un champ unique (un moteur de recherche, par exemple), le regroupement des champs n'est pas obligatoire.

Légende d'image

Lorsqu'un texte, adjacent à une image, contient des informations sur l'image (par exemple un copyright, une date, un auteur...) ou est destiné à compléter les informations apportées par l'image (par exemple un texte associé à une image dans une galerie d'images), on parle de légende d'image.

Lorsqu'une image est légendée il est nécessaire d'associer la légende de l'image à l'image par une relation de structure, de telle sorte que les technologies d'assistance puissent traiter l'image et sa légende comme un ensemble unique.

HTML5 propose d'associer une légende à une image via les éléments `figure` (l'ensemble de l'image et la légende) et `figcaption` (la légende).

Une image sans légende peut définir :

- Une image qui n'est pas insérée dans un élément `figure` ;
- Une image insérée dans un élément `figure` sans élément `figcaption`.

Note : lorsque le texte adjacent à l'image peut faire office de texte de remplacement, il n'est pas obligatoire de recourir à l'ensemble `figure`, `figcaption`, l'image pouvant être simplement traitée comme une image de décoration.

Vous pouvez consulter, à ce sujet, la note en anglais [Requirements for providing text to act as an alternative for images](#) du W3C.

Le nom, le rôle, la valeur, le paramétrage et les changements d'états

Un composant doit avoir un rôle et un nom appropriés. Ses valeurs, états et paramètres éventuels doivent également être accessibles et correctement transmis aux APIs d'accessibilité notamment.

Un composant peut s'appuyer sur un élément interactif HTML ou sur un élément non interactif surchargé par WAI-ARIA via un rôle ad hoc. Important : les boutons (balises `<button>` ou `<input type="button">`) lorsqu'ils sont contrôlés via JavaScript sont à évaluer avec le [critère 7.1](#).

Le nom peut être l'intitulé du composant (l'intitulé d'un bouton, par exemple).

La valeur est, par exemple, l'élément sélectionné d'une liste déroulante ou la valeur actuelle d'un curseur (`slider`).

Le rôle correspond au type d'élément défini par la spécification HTML ou WAI-ARIA (comme la balise `<button>` ou l'attribut WAI-ARIA `role="button"`).

Le paramétrage correspond aux informations particulières d'un composant, généralement mis à disposition par WAI-ARIA. Par exemple `aria-controls` est un paramètre qui transmet aux APIs l'information que le composant contrôle tel ou tel contenu (référéncé par son identifiant – attribut `id`).

Les changements d'état sont également mis à disposition par WAI-ARIA. Par exemple `aria-expanded` est un état permettant de signaler aux APIs que le composant est « ouvert » ou « fermé ». À noter

qu'un état peut également être transmis via le nom, lorsque l'intitulé est changé dynamiquement pour correspondre à l'état de la zone contrôlée notamment.

Ces paramètres ne sont pas obligatoires mais peuvent être requis s'ils sont indispensables pour rendre le composant accessible. C'est à l'auditeur de considérer les cas où ces paramètres sont indispensables en fonction du contexte lié à l'utilisation du composant.

L'auditeur doit également vérifier que, lorsqu'ils sont présents, ces paramètres sont correctement utilisés.

Pour ce faire (s'il juge cela pertinent compte tenu du contexte d'implémentation des composants et des choix ergonomiques mis en œuvre) il peut s'appuyer sur les recommandations d'utilisation de WAI-ARIA pour les composants ayant des attributs WAI-ARIA correspondant à un motif de conception tel que décrit dans le document [WAI-ARIA 1.1 Authoring Practices](#).

Note : les rôles, propriétés et états WAI-ARIA s'implémentent via des attributs, par exemple `role="banner", aria-hidden="true"`.

Lien

En HTML :

- Balise `<a>` possédant un attribut `href` ;
- Balise possédant un attribut WAI-ARIA `role="link"` et dont l'action de navigation est prise en charge par un script.

En SVG :

- Balise `<a>` possédant un attribut `xlink:href` en SVG 1.1.

Lien composite

En HTML, [lien](#) contenant à la fois du texte et un ou plusieurs enfants de type image :

- Image (balise `` ou balise ouvrante ayant l'attribut WAI-ARIA `role="img"`) ;
- Zone cliquable (balise `<area>`) possédant un attribut `href` ;
- Image objet (balise `<object>`) ;
- Image bitmap (balise `<canvas>`) ;
- Image vectorielle (balise `<svg>`).

Note importante : il est rappelé que l'utilisation de deux liens adjacents (lien image et lien texte) et identiques constitue une gêne importante pour l'utilisateur. Même si cela ne constitue pas une non-conformité, cet usage devrait être évité. Une manière de traiter ce type de liens est d'inclure l'image dans le lien texte de façon à constituer un lien composite, ce qui évitera la redondance.

Vous pouvez consulter à ce sujet la technique [H2 : Combining adjacent image and text links for the same resource](#).

Lien dont la nature n'est pas évidente

Lien qui peut être confondu avec un texte normal lorsqu'il est signalé uniquement par la couleur par certains types d'utilisateurs ne percevant pas ou mal les couleurs. Par exemple, dans ce texte

“Nouvelle grève à la SNCF”, si le mot “grève” est un lien signalé uniquement par la couleur, sa nature peut être ignorée par les utilisateurs ne percevant pas la couleur et accédant au contenu CSS activé.

Note : “signalés uniquement par la couleur” signifie que le lien n’est accompagné d’aucun marqueur visuel (icône, soulignement, bordure...). En conséquence, un lien de la même couleur que le texte environnant est concerné par ce critère.

Liens image

En HTML, [lien](#) contenant uniquement un ou plusieurs enfants de type image :

- Image (balise `` ou balise ouvrante ayant l’attribut WAI-ARIA `role="img"`);
- Zone cliquable (balise `<area>`) possédant un attribut `href` ;
- Image objet (balise `<object>`) ;
- Image bitmap (balise `<canvas>`) ;
- Image vectorielle (balise `<svg>`).

Liens ou boutons adjacents

Lien ou bouton présenté de manière adjacente à un élément dans la page. Le lien ou bouton doit être adjacent visuellement dans la représentation graphique (CSS activé) et dans le code HTML. Dans le code HTML, le lien ou bouton doit se situer juste avant ou juste après l’élément auquel il est adjacent.

Liens d’évitement ou d’accès rapide

Liens dont la fonction est de permettre de naviguer à l’intérieur de la page (lien d’évitement, lien d’accès au formulaire de recherche ou au menu...). Ces liens peuvent soit permettre d’accéder à une zone de la page (lien d’accès rapide) ou de sauter une zone dans la page (lien d’évitement).

Note 1 : Un lien d’évitement ou d’accès rapide fonctionnel est un lien dont l’activation permet de reprendre la lecture et la navigation clavier à partir de la cible du lien lors de l’utilisation des navigateurs et des aides techniques retenus dans l’environnement de test (ou « base de référence ») de l’audit.

Note 2 : les liens d’évitements ou d’accès rapide doivent être situés à la même place dans la présentation et dans le même ordre relatif dans le code source afin de satisfaire au [critère 12.2](#).

Liens SVG

Lien contenu dans un élément `<svg>`.

Dans le cas d’un lien SVG (version 1.1), le « nom accessible » est obtenu comme suit :

- Passage de texte associé par l’attribut WAI-ARIA `aria-labelledby` ;
- Sinon, contenu de l’attribut WAI-ARIA `aria-label` ;
- Sinon, contenu de l’élément `<title>` enfant direct du lien ;
- Sinon, contenu de l’attribut `xlink:title` ;
- Sinon, contenu texte d’un ou plusieurs éléments `<text>`.

Il faut cependant être vigilant car cet algorithme de calcul n'est pas encore pris en compte et effectif au sein des différents lecteurs d'écran. À ce jour, le support est disponible avec VoiceOver, mais incomplet ou lacunaire avec JAWS et NVDA. Si bien que le plus petit dénominateur commun sur lequel il est possible de se reposer pour fournir un intitulé au lien est l'élément `<text>`.

Liens de texte

En HTML, [lien](#) ne contenant aucun élément enfant de type :

- Image (balise `` ou balise ouvrante ayant l'attribut `WAI-ARIA role="img"`);
- Image objet (balise `<object>`);
- Image bitmap (balise `<canvas>`);
- Image vectorielle (balise `<svg>`).

Liste des valeurs possibles pour l'attribut `autocomplete`

[La liste des valeurs](#) disponibles est fournie par la spécification WCAG 2.1 :

- `name` – Nom complet ;
- `honorific-prefix` – Abréviation, civilité ou titre ;
- `given-name` – Prénom ;
- `additional-name` – Prénoms additionnels ;
- `family-name` – Nom de famille ;
- `honorific-suffix` – Suffixe honorifique ;
- `nickname` – Surnom, diminutif ;
- `organization-title` – Fonction, intitulé de poste ;
- `username` – Nom d'utilisateur ;
- `new-password` – Nouveau mot de passe (par exemple, lors de la création d'un compte ou d'un changement de mot de passe) ;
- `current-password` – Mot de passe actuel pour le compte identifié par le champ `username` (par exemple, lors d'une connexion) ;
- `organization` – Nom de l'organisation correspondant à la personne, à l'adresse ou à l'information de contact dans les autres champs associés avec ce champ ;
- `street-address` – Adresse postale (multiligne, nouvelles lignes conservées) ;
- `address-line1` – Adresse postale (une ligne par champ, ligne 1) ;
- `address-line2` – Adresse postale (une ligne par champ, ligne 2) ;
- `address-line3` – Adresse postale (une ligne par champ, ligne 3) ;
- `address-level4` – Le niveau administratif le plus détaillé, pour les adresses pourvues de quatre niveaux administratifs ;
- `address-level3` – Le troisième niveau administratif, pour les adresses pourvues d'au moins trois niveaux administratifs ;
- `address-level2` – Le deuxième niveau administratif, pour les adresses pourvues d'au moins deux niveaux administratifs ;
- `address-level1` – Le plus large niveau administratif d'une adresse, c'est-à-dire la province dans laquelle se trouve la localité ;
- `country` – Code pays ;
- `country-name` – Nom de pays ;

- `postal-code` – Code postal, code CEDEX (si le CEDEX est présent, ajouter "CEDEX", et ce qui le suit doit être ajouté dans le champ `address-level2`) ;
- `cc-name` – Nom complet figurant sur le moyen de paiement ;
- `cc-given-name` – Prénom figurant sur le moyen de paiement ;
- `cc-additional-name` – Prénoms additionnels figurant sur le moyen de paiement ;
- `cc-family-name` – Nom de famille figurant sur le moyen de paiement ;
- `cc-number` – Code identifiant le moyen de paiement (e.g., un numéro de carte bancaire) ;
- `cc-exp` – Date d'expiration du moyen de paiement ;
- `cc-exp-month` – Le mois de la date d'expiration du moyen de paiement ;
- `cc-exp-year` – L'année de la date d'expiration du moyen de paiement ;
- `cc-csc` – Code de sécurité du moyen de paiement (also known as the card security code (CSC), card validation code (CVC), card verification value (CVV), signature panel code (SPC), credit card ID (CCID), etc.) ;
- `cc-type` – Type de moyen de paiement (e.g. Visa) ;
- `transaction-currency` – La devise qui a la préférence de l'utilisateur lors d'une transaction ;
- `transaction-amount` – Le montant qui a la préférence de l'utilisateur lors d'une transaction (e.g., en réponse à une enchère ou à un prix soldé) ;
- `language` – Langue préférée ;
- `bday` – Date d'anniversaire ;
- `bday-day` – Le jour de la date d'anniversaire ;
- `bday-month` – Le mois de la date d'anniversaire ;
- `bday-year` – L'année de la date d'anniversaire ;
- `sex` – Identité de genre ;
- `url` – Page d'accueil ou une autre page Web correspondant à l'organisation, la personne, l'adresse ou à l'information de contact dans les autres champs associés avec ce champ ;
- `photo` – Photographie, icône ou une autre image correspondant à l'organisation, la personne, l'adresse ou à l'information de contact dans les autres champs associés avec ce champ ;
- `tel` – Numéro de téléphone complet, y compris le code pays ;
- `tel-country-code` – Code pays du numéro de téléphone ;
- `tel-national` – Numéro de téléphone sans la partie code pays, avec un préfixe interne au pays, s'il y a lieu ;
- `tel-area-code` – Indicatif régional du numéro de téléphone, avec un préfixe interne au pays, s'il y a lieu ;
- `tel-local` – Numéro de téléphone sans la partie code pays ni l'indicatif régional ;
- `tel-local-prefix` – La première partie du composant du numéro de téléphone qui suit l'indicatif régional, lorsque ce composant est scindé en deux parties ;
- `tel-local-suffix` – La seconde partie du composant du numéro de téléphone qui suit l'indicatif régional, lorsque ce composant est scindé en deux parties ;
- `tel-extension` – Numéro de téléphone d'un poste interne ;
- `email` – Adresse électronique ;
- `impp` – URL correspondant d'un protocole de messagerie instantanée (par exemple, "aim:goim?screenname=example" OU "xmpp:fred@example.net").

Suite d'éléments pouvant être regroupés sous la forme d'une liste structurée ordonnée, non ordonnée ou de définition. Par exemple la suite des liens d'un menu de navigation est une liste de liens non ordonnée, les différentes étapes d'une procédure sont une liste d'éléments ordonnés, le couple terme/description d'un glossaire est une liste de description. En HTML, les listes utilisent les balises suivantes :

- Liste ordonnée : balises `` et `` (chaque élément de la liste est affecté d'un marqueur indexé) ;
- Liste non ordonnée : balises `` et `` (chaque élément de la liste est affecté d'un marqueur non-indexé) ;
- Liste de description : balises `<dl>`, `<dt>` (terme à décrire) et `<dd>` (description).

Note 1 : En HTML5, la balise `<dl>` ne représente plus seulement une liste de définition, mais de manière générique toute liste de description qui peut comprendre comme groupe de termes-descriptions des noms et des définitions, des questions et réponses, des catégories et des sujets, etc.

Note 2 : Il est également possible de structurer les listes à l'aide des attributs WAI-ARIA `role="list"` et `role="listitem"` pour les listes ordonnées et non ordonnées.

Note 3 : la notion de « regroupés visuellement sous forme de liste » se caractérise par :

- La présence d'un marqueur visuel permettant de faire comprendre qu'il s'agit d'une liste non ordonnée par exemple -, •, *, etc. ;
- La présence d'un marqueur visuel permettant de faire comprendre qu'il s'agit d'une liste ordonnée par exemple un chiffre, une lettre grecque, etc. ;
- La présence d'une série d'éléments se suivant visuellement les uns les autres, avec une forme visuelle, une nature et un fonctionnement identique, mais sans avoir directement de marqueur visuel de liste (non ordonnée ou ordonnée), par exemple un menu de navigation.

Attention cependant toutes les listes ne nécessitent pas obligatoirement une structure de liste, par exemple une série de termes séparés par une virgule.

2.3.10 M

| Mécanisme de remplacement

Mécanisme généralement basé sur CSS, permettant à l'utilisateur de remplacer du texte par du texte en image et inversement sur le principe du style switcher. Le mécanisme peut utiliser un langage de script côté serveur ou un langage de script côté client.

| Mécanisme qui permet d'afficher un rapport de contraste conforme

Composant d'interface dont l'activation permet de modifier l'apparence du site ou de la page de manière à afficher les contenus avec un ratio de contraste suffisant. Le design de ce composant d'interface devra être conforme au [critère 3.2](#) et au [critère 3.3](#) sans avoir recours lui-même à un mécanisme permettant d'afficher un rapport de contraste conforme. Ce mécanisme doit conserver à l'identique les contenus et les fonctionnalités du site ou de la page qu'il modifie.

| Média non temporel

Contenu qui ne se déroule pas dans le temps, consultable via un plugin (Flash, Java, Silverlight...) ou via les éléments `svg` et `canvas` ; par exemple, une carte interactive en Flash, une application Flash ou

Java, un diaporama sont des médias non temporels. Un média non temporel peut contenir des médias temporels (un lecteur Flash qui propose une liste de vidéos à consulter, par exemple).

Note : l'utilisation du paramètre `wmode` pour un objet Flash avec les valeurs "transparent" et "opaque" invalide de fait le [critère 4.13](#). En effet, l'utilisation de ces valeurs a pour conséquence que l'animation Flash vue du côté des utilisateurs de lecteur d'écran est invisible.

Media temporel (type son, vidéo et synchronisé)

- Média temporel seulement audio : contenu sonore (Wav, Mp3...);
- Média temporel seulement vidéo : images ou photos en mouvement ou en séquence ;
- Média temporel synchronisé : flux audio ou vidéo synchronisé avec un autre format pour présenter de l'information et/ou comportant des composants temporels interactifs. Un média temporel peut être consulté de 2 manières différentes :
- Fichier à télécharger consultable avec un logiciel externe à la page web ;
- Contenu embarqué dans la page web et consultable dans la page web via :
 - Un plugin (par exemple une vidéo diffusée par un lecteur Flash) ;
 - L'élément `<video>` (par exemple une vidéo) ;
 - L'élément `<audio>` (par exemple un podcast) ;
 - L'élément `<svg>` (par exemple un dessin animé vectoriel) ;
 - L'élément `<canvas>` (par exemple un dessin animé en image bitmap) ;
 - L'élément `<bgsound>` pour diffuser un arrière-plan sonore à la page web.

Un média temporel peut être diffusé en temps réel ou être proposé en lecture de manière asynchrone (média pré-enregistré).

Note 1 : l'utilisation du paramètre `wmode` pour un objet Flash avec les valeurs "transparent" et "opaque" invalide de fait le [critère 4.13](#). En effet, l'utilisation de ces valeurs a pour conséquence que l'animation Flash vue du côté des utilisateurs de lecteur d'écran est invisible.

Note 2 : les gif animés, les animations d'images réalisées par JavaScript ou CSS ne sont pas considérés comme étant des médias temporels.

Note 3 : l'élément `<bgsound>` est spécifique à Internet Explorer et ne devrait pas être utilisé.

Menu et barre de navigation

Liste de liens permettant une navigation spécifique dans le site, dans une rubrique ou dans une collection de pages.

Les principales barres de navigation ([critère 12.2](#)) sont :

- Un menu de navigation ;
- Un fil d'ariane ;
- Une liste de navigation d'une liste de résultats ;

- Des liens d'évitement.

Il existe différents types de menu de navigation ([critère 12.1](#) et [critère 12.2](#)) :

- Menu de navigation principal ;
- Menu de sous-rubrique ;
- Menu contextuel ;
- Table des matières concernant un ensemble de pages.

Note : Les liens de pied de page renvoyant vers les mentions légales, plan du site et autres informations concernant le site ne sont pas considérés comme un menu de navigation principal.

Message de statut

Un message de statut informe l'utilisateur d'un changement de contenu dans la page sans interrompre son activité principale (il n'y a pas de changement de contexte par exemple un repositionnement du focus sur le message).

Un message de statut peut informer sur :

- Le succès ou le résultat d'une action ;
- L'état occupé d'une application ;
- L'état de progression d'un processus ;
- L'existence d'erreur.

Modifier ou annuler les données et les actions effectués

Procédés par lesquels un utilisateur peut modifier les données qu'il a saisies, faire annuler sa saisie ou faire annuler les actions découlant de sa saisie par exemple annuler une commande ou un virement bancaire.

Note : La page contenant un formulaire qui modifie ou supprime des données, ou qui transmet des réponses à un test ou un examen, ou dont la validation a des conséquences financières ou juridiques, doit indiquer explicitement la durée pendant laquelle l'utilisateur peut demander l'annulation de sa saisie. Elle devra également contenir la procédure à effectuer pour annuler cette saisie. Cette procédure n'a pas à être obligatoirement réalisable en ligne même si cela reste recommandé.

Moteur de recherche (interne à un site web)

Zone donnant accès directement (formulaire) ou indirectement (élément d'interface donnant accès au formulaire) au moteur de recherche qui permet d'effectuer des recherches sur les contenus de l'ensemble du site.

Note : Attention à ne pas confondre cette zone de recherche, unique dans le site, avec tout autre moteur de recherche permettant par exemple de faire des recherches sur une partie restreinte du site : un catalogue, les offres sur une section marchés publics...

Voir la définition technique de zone d'en-tête fournie par WAI-ARIA [search\(role\)](#).

Motif de conception

Un motif de conception (Design Pattern) est un modèle défini dans le document [WAI-ARIA 1.1 Authoring Practices](#) qui décrit la structure, les rôles et propriétés et le comportement clavier que doit respecter un composant JavaScript (widget).

Il est recommandé que les composants développés en JavaScript utilisant des attributs WAI-ARIA correspondant à un motif de conception respectent celui-ci. Attention cependant, les motifs de conception ne sont pas tous adaptés à un usage non applicatif, en particulier pour les sites proposant un affichage en contexte mobile.

Note 1 : compte tenu du manque de support de certaines propriétés et de certains rôles WAI-ARIA et de la grande variabilité des situations dans lesquelles un composant JavaScript peut être proposé, il est possible d'adapter des motifs de conception à des contextes ou des utilisations particulières.

Dans ce cas, le motif de conception adapté doit :

- Respecter la structure générale : par exemple un ensemble de panneaux (rôle WAI-ARIA `tabpanel`) d'un système d'onglets est forcément lié à un ensemble d'onglets (rôle WAI-ARIA `tablist`);
- Utiliser en remplacement d'un rôle ou d'une propriété WAI-ARIA mal supporté, un rôle ou une propriété WAI-ARIA équivalent, offrant un comportement et une restitution similaire.

Note 2 : Le fait d'enrichir un motif de conception de rôles ou propriétés WAI-ARIA supplémentaires dont la compatibilité avec l'accessibilité est contrôlée par le test de restitution sur l'environnement de test (ou « base de référence ») ne constitue pas une adaptation d'un motif de conception. Par exemple l'ajout de l'attribut WAI-ARIA `aria-hidden` sur les panneaux (rôle WAI-ARIA `tabpanel`) d'un système d'onglets ne définit pas un motif de conception adapté.

2.3.11 O

| Ordre de tabulation

Ordre dans lequel le focus se déplace (vers un élément suivant ou vers un élément précédent). L'ordre naturel est celui qui est implémenté via le code source. Lorsqu'il est modifié par l'utilisation de l'attribut `tabindex` ou par l'utilisation d'une commande JavaScript, c'est l'ordre modifié qui fait référence.

Attention : lorsqu'un élément initie un changement dans la page (changement de contexte, gestion de zones cachées, ajout de contenu, gestion de champs de formulaire...) il est nécessaire d'activer l'élément qui initie le changement pour tester la cohérence de l'ordre de tabulation.

2.3.12 P

| Page « plan du site »

Page dédiée présentant l'arborescence d'un site web, généralement sous forme de listes de liens organisées en rubriques et sous-rubriques donnant accès à l'ensemble des pages du site.

Note 1 : les liens du plan du site peuvent être constitués de balises `<a>` ou de balises `<area>`.

Note 2 : il n'est pas nécessaire que le plan du site contienne les liens vers toutes les pages du site, en revanche il est nécessaire qu'à partir du plan du site, l'utilisateur puisse atteindre l'ensemble des pages du site.

Passage de texte lié par `aria-labelledby` OU `aria-describedby`

Il s'agit d'un ou de plusieurs passages de texte identifiés par des `id` dont la valeur est unique dans la page et associés à un élément (champ de formulaire, bouton, etc.) par les attributs WAI-ARIA `aria-labelledby` OU `aria-describedby` sur le modèle suivant : `aria-labelledby="ID1 ID2 ID3..."` où la valeur de l'attribut utilisé est égale à la liste des valeurs d'attributs `id` des passages de texte à associer présents dans la page.

Note 1 : pour assurer une compatibilité maximum avec les agents utilisateurs, notamment Internet Explorer 11, il est recommandé d'implémenter un `tabindex="-1"` sur les passages de textes qui ne sont pas des éléments interactifs (bouton, liens, éléments de formulaires, etc.).

Note 2 : la valeur des attributs WAI-ARIA `aria-labelledby` OU `aria-describedby` ne doivent pas créer de référence récursive (A référence B qui référence A) ou traversante (A qui référence B qui référence C).

Présentation de l'information

Restitution visuelle des contenus via un navigateur en mode graphique. La présentation concerne le style, la position et les dimensions des éléments HTML et de leur contenu. La présentation de l'information doit être réalisée via CSS. Les éléments (`basefont`, `big`, `blink`, `center`, `font`, `marquee`, `s`, `strike` et `tt`) et les attributs (`align`, `alink`, `background`, `bgcolor`, `border`, `cellpadding`, `cellspacing`, `char`, `charoff`, `clear`, `compact`, `color`, `frameborder`, `hspace`, `link`, `marginheight`, `marginwidth`, `text`, `valign`, `vlink`, `vspace`) sont interdits.

Notes :

- Les attributs `width` et `height` utilisés sur d'autres éléments que les éléments ``, `<object>`, `<embed>`, `<canvas>` et `<svg>` sont également interdits ;
- L'attribut `size` utilisé sur d'autres éléments que l'élément `<select>` est également interdit ;
- L'élément `<u>` est interdit uniquement si le DOCTYPE du document ne correspond pas à HTML 5.

Pressé ou posé

Correspond aux gestionnaires d'événement JavaScript considérés comme des événements descendants (`mousedown`, `touchstart` par exemple).

Prise de focus

La prise de focus est l'état renvoyé par un élément qui reçoit l'attention suite à une action de l'utilisateur. Il y a trois moyens en HTML de donner le focus à un élément :

- En activant l'élément par un dispositif de pointage (exemple : souris) ;
- En atteignant l'élément par la touche tabulation ou majuscule + tabulation ;
- En activant l'élément par un raccourci clavier (`accesskey`).

Certains éléments reçoivent naturellement le focus, par exemple : `<a href>`, `<area href>`, `<button>`, `<input>`, `<object>`, `<select>`, `<label>`, `<legend>`, `<optgroup>`, `<option>` et `<textarea>`. Le comportement de l'élément, lors de la prise de focus, dépend de sa nature ; un lien, par exemple, devra être activé après la prise de focus (sauf utilisation de script). En revanche, un élément de formulaire, comme `<textarea>`, devra autoriser la saisie suite à la prise de focus. Les éléments `<label>` et `<legend>` ne reçoivent la prise de focus que via le pointeur souris. Pour l'élément `<label>`, le comportement attendu est de transférer la prise de focus sur l'élément qui lui est associé.

Note 1 : la spécification WAI-ARIA étend le rôle attribué à l'attribut `tabindex` en définissant que tout élément HTML peut acquérir la possibilité de recevoir le focus en lui attribuant la valeur `tabindex="0"`. En revanche, aucun comportement n'est attribué via la seule présence de `tabindex`. De même, la valeur `tabindex="-1"` lorsqu'elle est utilisée sur un élément recevant naturellement le focus retire l'élément qui en est affecté du plan de tabulation en inhibant sa capacité à signaler la « prise de focus ». L'utilisation de `tabindex`, conformément à la spécification WAI-ARIA, peut valider certains tests relatifs à la gestion du focus de tabulation, notamment.

Note 2 : l'indication visuelle du focus du navigateur ne doit pas être supprimée ou dégradée sauf si un style du focus défini par l'auteur est visible et suffisamment contrasté au regard du [critère 3.3](#).

Procédé de rafraîchissement

Technique visant à modifier le contenu d'un ou de plusieurs éléments de la page web. Le procédé de rafraîchissement peut s'effectuer par rechargement automatique de la page ou de manière dynamique sans rechargement de la page (via AJAX, par exemple). L'utilisateur doit pouvoir contrôler chaque procédé de rafraîchissement de manière indépendante.

Propriété CSS déterminant une couleur

Cela concerne les propriétés suivantes : `color`, `background-color`, `background`, `border-color`, `border`, `outline-color`, `outline`.

Note : l'utilisation d'une image de fond pour insérer une couleur (propriété `background:url(...)`) est également concernée.

2.3.13 R

Raccourci clavier

Un moyen de déclencher une action associée à un composant de l'interface utilisateur en appuyant sur une ou plusieurs touches.

Note : les « Access keys » (attribut HTML `accesskey`) sont bien des raccourcis clavier, mais ils ne sont pas concernés par le [critère 12.10](#) dans la mesure où leur activation nécessite déjà l'usage de touches de modification (variables suivant les navigateurs).

Redirection

Procédé qui consiste pour l'affichage d'une page sur le poste client à rediriger l'utilisateur vers une autre page, sur le même domaine ou sur un domaine différent.

Règles d'écriture

Le code source doit respecter les règles suivantes en accord avec la déclaration de type de document utilisée dans la page :

- Pas de balise ouvrante ou fermante sans `<` ou `>` (exemple d'erreur : `li>vélo`);
- pas de balise fermante avec `/` manquant (exemple d'erreur : `vélo`);
- pas de valeur d'attribut avec des `"` ou ``` manquant (exemple d'erreur : `alt="vélo`);
- pas de valeurs multiples d'attribut séparées par un espace sans `"` ou ``` (exemple d'erreur : `alt=mon beau vélo`);
- pas d'espace manquant entre les attributs (exemple : `alt="vélo" title="vélo"`);
- pas de balise fermante manquante pour les éléments qui en exigent une (exemple d'erreur : `<object> sans </object>`).

Relâché ou relevé

Correspond aux gestionnaires d'événement JavaScript considérés comme des événements ascendants (`mouseup`, `touchend` par exemple).

Résumé (de tableau)

Un résumé est un passage de texte associé à un tableau de données complexe. Il permet de donner des informations sur la nature et la structure du tableau afin d'en faciliter l'utilisation par les utilisateurs de technologies d'assistance par exemple.

Note : en HTML5, la seule technique utilisable actuellement est celle qui consiste à insérer le résumé directement dans le titre (balise `<caption>`) en masquant le résumé via CSS si nécessaire.

Dans les versions précédentes de HTML, le résumé peut être inséré via un attribut `summary` sur la balise `<table>`.

Dans le cas d'une balise avec l'attribut WAI-ARIA `role="table"`, le résumé doit être fourni au moyen d'un attribut `aria-describedby` et être correctement restitué par les technologies d'assistance.

2.3.14 S

Script

Code généralement écrit sous forme d'une liste de commandes (par exemple JavaScript). Les langages interprétés côté client nécessitent un navigateur compatible sur lequel l'exécution du langage est active. Les commandes d'un langage de script côté client peuvent être embarquées ou contenues dans un fichier externe. Dans les deux cas, l'insertion se fait via la balise `<script>`.

Sens de lecture

Indique le sens de lecture du document ou d'un passage de texte via l'attribut `dir`, `dir="ltr"`, par exemple. Les deux valeurs reconnues sont :

- `ltr` (left to right) indique un sens de lecture de gauche à droite ;
- `rtl` (right to left) indique un sens de lecture de droite à gauche.

Note : en l'absence d'indication de sens de lecture via l'attribut `dir` sur l'élément `html`, `body`, ou un des parents du texte analysé, le sens de lecture par défaut est de gauche à droite (valeur `ltr`).

Site web

Ensemble de toutes les pages web :

- Reliées par des liens web ;
- Appartenant au même nom de domaine (ex : `design.numerique.gouv.fr`) ;
- Qui constituent un ensemble cohérent du point de vue de l'utilisateur.

Cas particulier des pages web d'un sous-domaine ; un sous-domaine peut :

- Soit appartenir au site web attaché au nom de domaine, si l'utilisateur en a une perception cohérente avec les autres pages du site web (par exemple : même structure, même navigation...);
- Soit ne pas appartenir au site web attaché au nom de domaine (par exemple : différents blogs en sous-domaine d'un nom de domaine et sans relation les uns avec les autres).

Sous-titres synchronisés (objet multimédia)

Texte des informations audio (paroles d'un personnage, bruit important pour comprendre l'action...) présentes dans un média temporel et affiché de manière synchrone avec le flux de l'objet multimédia.

Note 1 : pour différencier les sources sonores (différents personnages, voix off...), il est recommandé d'utiliser un mécanisme approprié (mise entre crochets, mise en italique, annonce explicite du type "voix off : ...").

Note 2 : il ne faut pas confondre le sous-titrage pour la traduction (attribut `kind="subtitles"` de la balise `<track>` en HTML5, par exemple) et le sous-titrage pour sourds et malentendants (attribut `kind="captions"` de la balise `<track>` en HTML5, par exemple). Ces deux types de sous-titrage poursuivent des buts différents. Seule la présence et la pertinence d'un sous-titrage pour sourds et malentendants permet d'être conforme.

Structure du document

Ensemble d'éléments permettant de définir les grandes zones d'une page HTML telles que la zone d'en-tête de la page, les zones de navigation principale et secondaire, la zone de contenu principal et la zone de pied de page.

Système de navigation

Tout procédé permettant une navigation dans le site ou dans une page, les systèmes de navigation retenus sont :

- Menu de navigation principal ;
- Table des matières ;
- Plan du site ;
- Moteur de recherche.

2.3.15 T

Tableau de données

Un tableau de données est une structure introduite par une balise `<table>` ou lorsqu'il est correctement restitué par les technologies d'assistance par une balise pourvue d'un attribut WAI-ARIA `role="table"`. Cette balise permet de structurer des informations en lignes et en colonnes via des cellules de données et des cellules d'en-têtes.

Tableau de données ayant un titre

Tableau de données ayant un attribut ou contenant une balise dont le contenu fait office de titre.

Tableau de données précédé ou suivi d'un passage de texte associé au tableau faisant office de titre.

Dans la mesure où il est bien correctement restitué et associé par les technologies d'assistance au tableau de données, le titre associé peut être :

- Dans une balise `<caption>` ;
- Dans un attribut `title` ;
- Dans un attribut WAI-ARIA `aria-label` ;
- Dans une balise associée au tableau de données via un attribut WAI-ARIA `aria-labelledby` sur le tableau.

Note : seule la balise `<caption>` étant complètement supporté par l'ensemble des technologies d'assistance, il est fortement recommandé de privilégier cette solution lors de la mise en œuvre afin d'éviter de nombreuses vérifications dans les différentes combinaisons prévues par l'environnement de test (ou « base de référence »).

Tableau de données complexe

Un tableau de données est une structure introduite par une balise `<table>` ou lorsqu'il est correctement restitué par les technologies d'assistance par une balise pourvue d'un attribut WAI-ARIA `role="table"`.

Lorsqu'un tableau de données contient des en-têtes qui ne sont pas répartis uniquement sur la première ligne et/ou la première colonne de la grille ou dont la portée n'est pas valable pour l'ensemble de la colonne ou de la ligne, on parle de tableau de données complexe. Il est alors nécessaire de fournir un « résumé » permettant d'en expliquer sa nature et sa structure afin d'en faciliter la consultation pour des utilisateurs de technologies d'assistance par exemple.

Tableau de mise en forme

Technique qui utilise un élément HTML (balise `<table>`) pour contrôler l'affichage d'informations via des cellules (balise `<td>`).

Taille des caractères

Valeur attribuée aux polices de caractères présentes sur une page web.

Texte stylé

Texte dont la mise en forme est contrôlée par une feuille de styles.

Titre

Élément HTML (balise `h`) à 6 niveaux de hiérarchie (de `h1` pour le titre le plus important à `h6` pour le moins important) ou élément HTML ayant les attributs WAI-ARIA `role="heading"` et `aria-level` permettant de structurer l'information d'un contenu web.

Assurer une stricte hiérarchie entre les titres d'une page web est une bonne pratique, mais la présence de sauts hiérarchiques n'invalide pas le critère tant que cette hiérarchie plus lâche reste cohérente (un titre `<h3>` peut ainsi venir directement après un titre `<h1>`, par exemple). La hiérarchie de titres ne doit pas obligatoirement contenir un titre `<h1>`. Même si ces pratiques ne sont pas encouragées, elles n'invalident pas le critère.

Note : les titres visuellement cachés via CSS sont considérés comme présents et valident le [critère 9.1](#).

Titre de cadre

Contenu de l'attribut `title` de la balise `<iframe>` ou `<frame>` permettant de connaître la nature du contenu diffusé via le cadre lorsque l'utilisateur navigue de cadre en cadre ou affiche la liste des cadres de la page par exemple.

Note 1 : Certains cadres servent uniquement à des opérations techniques tels que des traitements applicatifs destinés à préparer ou piloter des contenus affichés dans la page comme les cadres utilisés par certains réseaux sociaux comme Facebook par exemple.

Si ces cadres sont dépourvus de titre de cadre fournis par le service distant, ou si les titres de cadres sont jugés non pertinents, des mentions génériques peuvent être utilisées, par exemple `title="contenus techniques Facebook"`.

Note 2 : Si cela ne gêne pas le fonctionnement de ce type de cadre, il est possible de les rendre indisponibles aux technologies d'assistance en utilisant l'attribut WAI-ARIA `aria-hidden="true"`. Dans ce cas le [critère 2.1](#) et le [critère 2.2](#) seront non applicables.

Titre de page

Contenu de la balise `<title>` d'une page web permettant d'identifier de manière claire, concise et unique les contenus/la nature de la page (« Plan du site `www.nomdusite.fr` » pour une page présentant le plan du site web, par exemple).

Transcription textuelle (média temporel)

Contenu textuel associé à un média temporel par la technique appropriée (texte codé en HTML ou dans un fichier texte qui se trouve dans la même page ou consultable suivant un lien). Ce contenu donne accès à l'utilisateur (de manière indépendante de la consultation de l'objet multimédia) à :

- La totalité de ce qui y est exprimé oralement ;
- Toutes les informations descriptives nécessaires à une compréhension équivalente de l'action.

Ces informations textuelles doivent être présentées dans l'ordre chronologique de leur apparition dans le média temporel.

Note : la transcription textuelle doit se situer à l'extérieur de la balise `<object>`.

| Type de document

Ensemble de données de référence qui permet aux agents utilisateurs de connaître les caractéristiques techniques des langages utilisés sur la page web (balise `doctype`).

| Type et format de données

Indication concernant le type et le format des données attendus lors de la saisie d'un champ de formulaire. Par exemple :

- Date (jj/mm/aaaa) ;
- Montant en euros ;
- Code postal (5 chiffres : ex. 75001).

Note importante : lorsque le type de champ de formulaire propose un masque de saisie, par exemple les champs `date` ou `time`, l'indication de format n'est pas nécessaire.

2.3.16 U

| Uniquement à des fins de présentation

Uniquement à des fins de présentation : utilisation de balises HTML pour une finalité différente de celle prévue dans les spécifications (au regard du type de document déclaré). Exemples : utilisation des balises `h` à seule fin de créer un effet typographique ; utilisation de la balise `<blockquote>` à seule fin de mettre un paragraphe en retrait, etc.

Note 1 : l'utilisation d'éléments `<div>` ou `` ou plusieurs `
` pour créer visuellement un paragraphe est considérée comme non conforme et invalide le critère.

Exemple : `<div>`, paragraphes d'un bloc de texte simulés, `
`, à l'aide de plusieurs balises `
`, `</div>`

Note 2 : WAI-ARIA propose un rôle `presentation` permettant de supprimer la sémantique d'un élément, par exemple `<h1 role="presentation">Titre</h1>`. Dans ce cas, le texte sera correctement restitué mais son rôle de titre ne le sera plus. L'utilisation du rôle `presentation` peut être requise lorsque l'on utilise un [motif de conception](#) WAI-ARIA.

Le rôle WAI-ARIA `presentation` peut être également utilisé pour supprimer la sémantique d'un élément lorsque ce dernier est utilisé uniquement à des fins de présentation, par exemple `<blockquote role="presentation">` aura le même effet qu'une absence d'élément `<blockquote>`.

Même si cette utilisation est fortement déconseillée (dans le cas de technologie d'assistance qui n'implémenteraient pas WAI-ARIA par exemple) elle peut être considérée comme conforme à WCAG. En revanche l'utilisation d'un rôle WAI-ARIA `presentation` sur un élément dont la nature (par exemple la sémantique) est essentielle à la compréhension du contenu est une violation des règles WCAG (particulièrement de l'échec [F92](#)) et invalide le critère.

2.3.17 V

| Version accessible (pour un document en téléchargement)

Les documents en téléchargement dont les types de format sont reconnus compatibles avec l'accessibilité doivent être rendus accessibles soit directement soit par l'intermédiaire d'une version accessible ou d'une version en HTML. Les formats de document dont la compatibilité est reconnue sont :

- Microsoft Office (Word 2003, OOXML) ;
- Open Office Org (ODF) ;
- Adobe PDF ;
- EPUB/Daisy.

Note : le format texte (txt) ne peut pas être utilisé pour produire une version accessible pour un document en téléchargement.

| Version alternative « audio seulement »

Une version « audio seulement » est une version sonore, sous la forme d'un simple fichier au format MP3 par exemple, utilisée comme alternative à une vidéo seulement (vidéo sans information sonore). Les seuls utilisateurs impactés par l'accessibilité étant les personnes aveugles, qui ne peuvent pas voir la vidéo, WCAG considère comme acceptable de proposer en alternative une version sonore.

La version « audio seulement » doit contenir toutes les informations visuelles importantes de la vidéo.

Généralement il est plus simple de produire une version sonore qu'une version textuelle lorsque la vidéo est très descriptive (la transcription textuelle nécessitant souvent un travail rédactionnel important). Il est rappelé, néanmoins, que seule la transcription textuelle assure un accès universel aux informations diffusées par la vidéo, dans le cas où un utilisateur ne serait pas en capacité de lancer la vidéo par exemple.

2.3.18Z

| Zone cliquable

Région d'une image réactive à laquelle une action a été associée ; par exemple, le déclenchement d'un événement en cliquant sur un lien (pour une zone cliquable côté client : balise `<area>` avec l'attribut `href`). Les balises `<area>` sont contenues dans la balise `<map>`.

Pour les images réactives côté serveur, les coordonnées sont détenues sur le serveur.

| Zone de contenu principal

Zone contenant les principaux contenus de la page, là où se trouvent les informations et fonctionnalités de fond (donc en dehors des menus, de la recherche ou des zones secondaires de publicités, actualités connexes...).

Note : Cette zone est unique dans la page. Elle peut être difficile à déterminer sur certaines pages particulières, comme la page d'accueil.

Voir la définition technique fournie par WAI-ARIA : [Main \(role\)](#).

| Zone d'en-tête

Zone située en haut du document et contenant généralement le titre du document, un logo, un slogan...

Note : Attention à ne pas confondre cette zone d'en-tête, unique dans le site, avec tout contenu pouvant être balisé en HTML5 avec l'élément `<header>`.

Voir la définition technique fournie par WAI-ARIA : [Banner \(role\)](#).

| Zone de pied de page

Il s'agit des informations concernant le fonctionnement du site ou les informations légales. On y trouve par exemple les mentions légales, les crédits, les conditions d'utilisation, le plan du site et éventuellement la page accessibilité.

Note : Attention à ne pas confondre cette zone de pied de page, unique dans le site, avec tout contenu pouvant être balisé en HTML5 avec l'élément `<footer>`.

Voir la définition technique fournie par WAI-ARIA : [Contentinfo \(role\)](#).

| Zone (d'une image réactive)

Zone cliquable ou zone non cliquable d'une image réactive côté client ou zone cliquable d'une image réactive côté serveur.

| Zone non cliquable

Région d'une image réactive à laquelle aucune action n'est associée. Une zone non cliquable côté client est contenue dans une balise `<area>` :

- Avec l'attribut `nohref` lorsque le code HTML de la page n'est pas du HTML5 ;
- Sans attribut `href` en HTML5.

Les balises `<area>` sont contenus dans la balise `<map>`.

2.4 Environnement de test (ou « base de référence ») pour la restitution des contenus web par les technologies d'assistance

Quelques critères RGAA, notamment ceux de la thématique JavaScript, incluent des tests de restitution à effectuer sur des technologies d'assistance associées à des navigateurs et des systèmes d'exploitation.

Les tests effectués selon ces combinaisons (technologie d'assistance, système d'exploitation, navigateur) permettent de déclarer qu'un dispositif HTML / WAI-ARIA est « [compatible avec l'accessibilité](#) » tel que défini par WCAG.

Les combinaisons ont été établies à partir de la liste des technologies d'assistance dont l'usage est suffisamment répandu, ou, dans certains cas lorsqu'elle est fournie de manière native et constitue le moyen privilégié d'accès à l'information et aux fonctionnalités.

2.4.1 Environnement de test Ordinateur (desktop)

Les systèmes d'exploitation retenus sont Windows et Mac OS X et les navigateurs, Internet Explorer, Firefox et Safari. Il appartient à l'auditeur de définir, en concertation avec les responsables du site audité, les versions de système d'exploitation et de navigateur en adéquation avec le contexte d'usage du site et l'environnement de test utilisé lors du développement du site. Les versions des technologies d'assistance à utiliser seront soit la dernière disponible en langue française sur le système d'exploitation retenu soit la version précédente

Lorsque le site ou l'application est destiné à un public dont l'équipement est maîtrisé, les tests devront se faire sur un environnement de test adapté au contexte de l'environnement maîtrisé.

Pour qu'un dispositif HTML / WAI-ARIA ou son alternative soit considéré comme compatible avec l'accessibilité, il faut qu'il soit pleinement fonctionnel, en termes de restitution et de fonctionnalités, sur au moins une des combinaisons suivantes.

Environnement de test Ordinateur (desktop) - Combinaison 1

NVDA (dernière version) précédente)	JAWS (version dernière version)	Firefox Firefox ou Internet Explorer	Safari
--	------------------------------------	---	--------

Environnement de test Ordinateur (desktop) - Combinaison 2

JAWS (version précédente)	Firefox
NVDA (dernière version)	Firefox ou Internet Explorer
VoiceOver (dernière version)	Safari

2.4.2 Environnement de test Terminal mobile

Les systèmes d'exploitation retenus sont Android et iOS et les navigateurs Chrome et Safari. Il appartient à l'auditeur de définir, en concertation avec les responsables du site audité, les versions de système d'exploitation et de navigateur en adéquation avec le contexte d'usage du site et l'environnement de test utilisé lors du développement du site. Les versions des technologies d'assistance à utiliser seront soit la dernière disponible en langue française sur le système d'exploitation retenu, soit la version précédente.

Pour tester un site web sur un terminal mobile, l'environnement de test devra comporter une des deux combinaisons complémentaires suivantes :

Environnement de test Terminal mobile - Combinaison 1

iOS	VoiceOver (dernière version)	Safari
-----	------------------------------	--------

Environnement de test Terminal mobile - Combinaison 2

Android natif	TalkBack (dernière version)	Chrome
---------------	-----------------------------	--------

À noter que dans le cas d'un site web mobile grand public, il est fortement conseillé de tester dans les deux environnements.

2.4.3 Autres environnements

Enfin, en fonction du contexte du site audité, d'autres technologies d'assistance complémentaires peuvent être utiles telles que :

- ZoomText sur Windows ou Mac OSX.
- Dragon Naturally Speaking sur Windows ou Mac OSX.

2.4.4 Environnement maîtrisé

Lorsque le site web est destiné à être diffusé et utilisé dans un environnement maîtrisé, l'environnement de test (base de référence) doit être constitué des configurations (technologie d'assistance, système d'exploitation, navigateur) effectivement utilisées dans l'environnement maîtrisé.

Par exemple, lorsque le site web est exclusivement diffusé dans un environnement GNU/Linux, les tests devront être réalisés uniquement sur les navigateurs et les technologies d'assistance utilisés par les agents sur cette plateforme. Cet environnement de test (base de référence) se substitue à l'environnement de test (base de référence) utilisé en environnement non maîtrisé.

2.5 Références documentaires

Le RGAA a été établi en utilisant un certain nombre de références et de sources documentaires. Ce document liste les références qui ont été utilisées.

Sauf mention contraire, tous les documents sont en anglais.

2.5.1 Documents de référence

- [Accessibility requirements for ICT products and services, EN 301 549 V2.1.2 \(.pdf\)](#) (2018-08) ;
- [Web Content Accessibility Guidelines \(WCAG\) 2.1](#) - référence à la date de juin 2018 ;
- [Règles pour l'accessibilité des contenus Web \(WCAG\) 2.1](#) - traduction Française Agréée;
- [Understanding WCAG 2.1](#) - référence à la date d'avril 2019 ;
- [Techniques for WCAG 2.1](#) - référence à la date d'avril 2019 ;
- [HTML 4.01 Specification](#) ;
- [HTML 5.2](#) - référence à la date de décembre 2017 ;
- [Accessible Rich Internet Applications \(WAI-ARIA\) 1.1](#) - référence à la date de décembre 2017 ;
- [WAI-ARIA Authoring Practices 1.1](#) - référence à la date de février 2019 ;
- [Using ARIA](#) - référence à la date de septembre 2018 ;
- [ARIA in HTML](#) - référence à la date d'avril 2019.

2.5.2 Autres documents

- [Web Content Accessibility Guidelines Working Group \(Wiki\)](#) ;
- [User Agent Accessibility Guidelines \(UAAG\) 2.0](#) - référence à la date de décembre 2015 ;
- [Authoring Tool Accessibility Guidelines \(ATAG\) 2.0](#) - référence à la date de septembre 2015 ;
- [WAI-ARIA 1.0 User Agent Implementation Guide](#) - référence à la date de mars 2014 ;
- [Accessible Name and Description Computation 1.1](#) - référence à la date d'avril 2019 ;
- [Core Accessibility API Mappings 1.1](#) - référence à la date de décembre 2017 ;
- [HTML Accessibility API Mappings 1.0](#) - référence à la date d'avril 2019 ;
- [SVG Accessibility API Mappings 1.0](#) - référence à la date d'avril 2019.

2.6 Licence

Ce document est placé sous [licence ouverte 2.0 ou ultérieure](#).

La partie 2 du RGAA 4.1 ou méthode technique de vérification de la conformité à la norme de référence est adaptée du référentiel technique RGAA 3 mise à jour 2017 (liste des critères, glossaire, cas particuliers, notes techniques, environnement de tests « base de référence ») qui est lui-même une copie adaptée du [référentiel AccessiWeb HTML5/ARIA](#) - Version de travail du 19/12/2013 - Édité par l'association BrailleNet.

