

GUIDE

Accessibilité Numérique

Comment favoriser l'accès au monde numérique pour tous

W3C

WCAG 2.1

WEB CONTENT ACCESSIBILITY
GUIDELINES

Avril 2023

RGAA 4.1

Référentiel général d'amélioration
de l'accessibilité

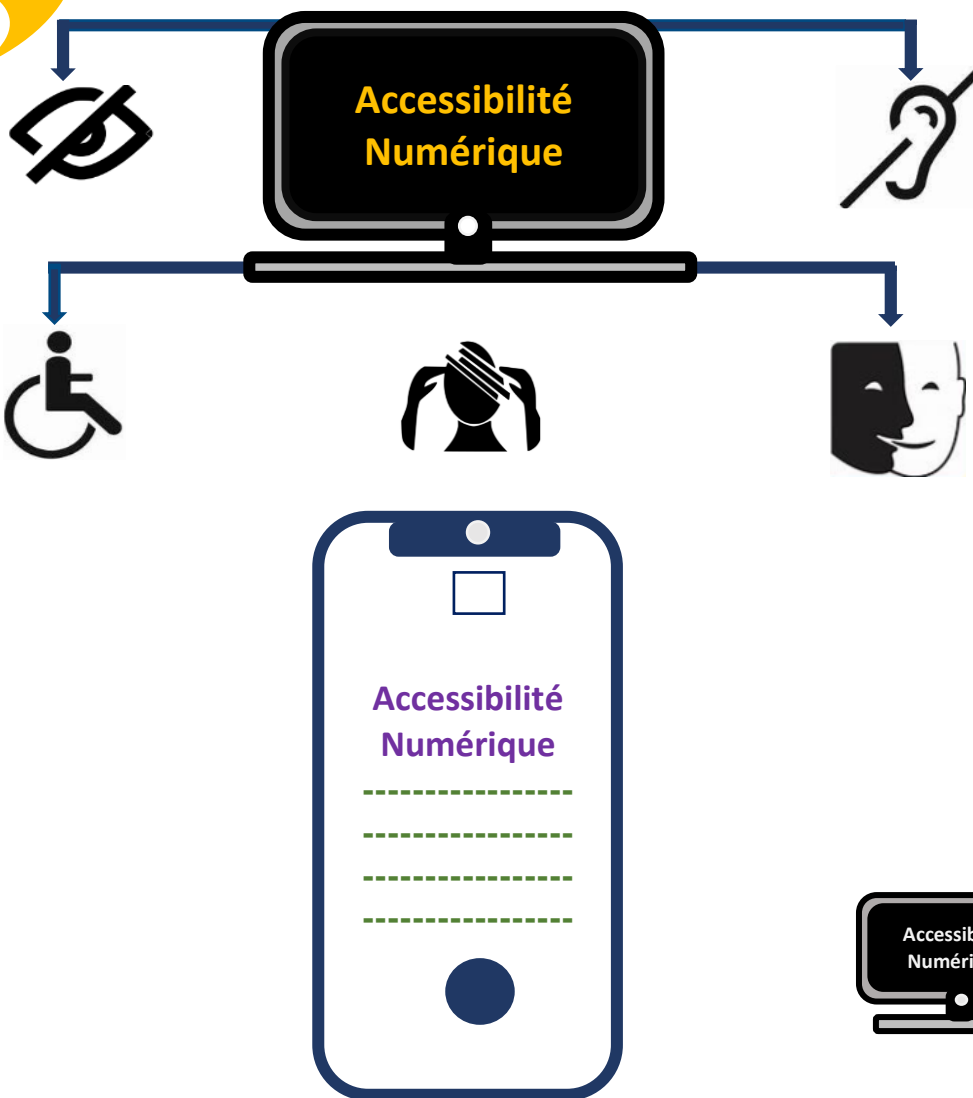
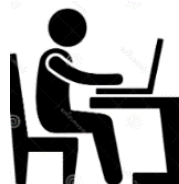


Table des matières

A qui s'adresse ce guide ?	3
1 Handicap et compensations	4
1.1 Les différents types de Handicap et les compensations	4
1.1.1 Handicap visuel	4
1.1.2 Handicap auditif	4
1.1.3 Handicap moteur	5
1.1.4 Handicap mental	5
1.1.5 Handicap psychique	5
1.1.6 Handicap cognitif	6
2 Handicap et accessibilité	7
2.1 Qu'est-ce que l'accessibilité numérique ?	8
2.2 WCAG et RGAA : référentiels d'accessibilité pour les sites web	8
2.2.1 Web Content Accessibility Guidelines (WCAG)	8
2.2.2 Référentiel d'amélioration de l'accessibilité (RGAA)	10
2.2.3 Niveaux de conformité du web	10
2.3 Conformité et validation du web	11
2.3.1 Outils de conformité du web	11
2.3.2 Outils de contraste sur le web	12
2.3.3 Outils de validation	13
2.3.4 Outils de tests manuels	13
2.4 Squelette de base pour une page web accessible	14
3 Les principes de base de l'accessibilité numérique	15
3.1 Clarté	15
3.1.1 Simplicité du design	15
3.1.2 Qualité du contenu	15
3.1.3 Structure et organisation	15
3.1.4 Accessibilité	16
3.2 Balise de titre « title » pertinente	16
3.3 Utiliser des balises de titres appropriées	16
3.4 Contenu alternatif	17
3.5 Contraste élevé	18
3.6 Navigation simple	20
3.6.1 Le menu principal	20
3.6.2 Les liens dans une page web	20
3.6.3 Les boutons dans une page web	22
3.6.4 La barre de recherche	23
3.7 Les filtres et les options de tri	24

3.8	Commandes de clavier.....	25
3.9	Sous-titres.....	27
3.10	Description audio.....	28
3.11	Taille de police réglable.....	28
3.12	Choix de la langue.....	30
3.13	Tableaux et libellés.....	31
3.14	Formulaire accessible.....	32
3.15	Restituer les composants interactifs.....	34
3.16	Séparation du fond et la forme.....	35
4	Accessibilité des documents bureautiques.....	36
4.1	Utiliser une structure de document cohérente.....	36
4.2	Utiliser des couleurs contrastées.....	36
4.3	Utiliser une police facile à lire.....	37
4.4	Ajouter des descriptions alternatives aux images.....	37
4.5	Utiliser les tableaux de manière appropriée.....	38
4.6	Utiliser les listes à puces.....	38
4.7	Ajouter des liens.....	39
4.8	Vérifier l'accessibilité du document Word.....	39
4.9	Accessibilité d'un document PDF.....	40
4.10	Vérifier l'accessibilité du PDF.....	40
4.11	Table des matières accessible.....	41
5	QCM sur l'accessibilité numérique.....	42
	Sites Internet de référence.....	44
	Des Ressources à consulter.....	44

A qui s'adresse ce guide ?

Ce guide s'adresse à toutes les personnes curieuses de mieux comprendre ce qu'est l'accessibilité numérique et qui souhaitent créer des contenus accessibles pour tous. Il présente les définitions essentielles à connaître, ainsi que quelques points clés sur le handicap et les outils de compensation.

Le guide fournit des informations sur les normes d'accessibilité numérique, les bonnes pratiques et les outils pour aider à concevoir et à développer des contenus numériques accessibles à tous les utilisateurs, y compris ceux ayant des handicaps. Il peut également être utile pour les responsables de l'accessibilité et les personnes impliquées dans la mise en œuvre et l'application des politiques d'accessibilité numérique, ainsi que pour les professionnels de la formation et de la sensibilisation à l'accessibilité numérique.

Pour pratiquer l'accessibilité numérique au quotidien, le guide propose 16 conseils pratiques illustrés. Il fournit des conseils et des illustrations pour rendre un document bureautique accessible. En suivant ces conseils, les contenus deviendront plus accessibles à une plus grande variété de personnes en situation de handicap.

Ce guide en accessibilité numérique est destiné à aider les professionnelles et les parties prenantes à intégrer l'accessibilité numérique dans leur travail quotidien, afin de créer des contenus numériques qui soient inclusifs et accessibles à tous les utilisateurs.

1 Handicap et compensations

Le handicap se réfère à toute limitation physique, mentale ou sensorielle qui entrave la capacité d'une personne à effectuer certaines tâches ou activités de la vie quotidienne. Le handicap peut être présent depuis la naissance ou survenir à la suite d'une maladie, d'un accident ou du vieillissement.

Les compensations quant à elles sont des stratégies ou des techniques utilisées pour aider une personne en situation de handicap à surmonter les obstacles ou à compenser les limitations qu'elle rencontre. Les compensations sont souvent nécessaires pour garantir l'égalité des chances pour les personnes handicapées dans tous les domaines de la vie, y compris l'emploi, l'éducation, le logement et l'accès aux services publics, l'accès aux services numériques.

Les compensations visent à permettre aux personnes en situation de handicap de participer pleinement à la vie sociale, professionnelle et personnelle. Elles peuvent contribuer à améliorer la qualité de vie de ces personnes en leur offrant plus d'autonomie et d'indépendance.

1.1 Les différents types de Handicap et les compensations

Il existe plusieurs types de handicaps, qui peuvent affecter différentes fonctions corporelles et mentales :

1.1.1 Handicap visuel



Le handicap visuel est un type de handicap qui affecte la capacité d'une personne à voir. Ce handicap peut inclure la cécité totale (perte complète de la vision) ou une déficience visuelle partielle (perte partielle de la vision). Les causes les plus courantes de déficience visuelle sont la dégénérescence maculaire liée à l'âge, le glaucome, la rétinopathie diabétique et la cataracte.

Les personnes en situation de handicap visuel peuvent avoir besoin d'aide pour accomplir des tâches quotidiennes telles que se déplacer dans des environnements inconnus, lire ou écrire des documents, utiliser des technologies, cuisiner et faire des achats en magasin ou en ligne. Les adaptations pour les personnes atteintes de handicap visuel peuvent inclure des aides visuelles telles que des cannes blanches ou des chiens-guides, des systèmes de lecture en braille, des logiciels de lecture d'écran pour ordinateurs, des livres audios et des adaptations de l'environnement de travail ou de la maison.

1.1.2 Handicap auditif



Le handicap auditif est un type de handicap qui affecte la capacité d'une personne à entendre. Ce handicap peut inclure la surdité totale (perte complète de l'audition) ou une déficience auditive partielle (perte partielle de l'audition). Les causes les plus courantes de déficience auditive sont le vieillissement, les infections de l'oreille, les traumatismes sonores et les maladies génétiques.

Les personnes en situation de handicap auditif peuvent avoir besoin d'aide pour communiquer avec les autres et pour accéder à des informations auditives telles que des conversations, des annonces publiques et des programmes télévisés.

Les adaptations pour les personnes atteintes de handicap auditif peuvent inclure des aides auditives telles que des prothèses auditives et des implants cochléaires, des interprètes en langue des signes, des systèmes de sous-titrage et des adaptations de l'environnement de travail ou de la maison.

1.1.3 Handicap moteur



Le handicap moteur est un type de handicap qui affecte la mobilité et la coordination d'une personne. Ce handicap peut être causé par une blessure, une maladie ou une condition congénitale. Il peut être présent dès la naissance ou se développer plus tard dans la vie, tout comme d'autres types de handicap. Les personnes en situation de handicap moteur peuvent rencontrer des difficultés à marcher, à se tenir debout, à saisir des objets ou à effectuer des tâches fines avec leurs mains.

Les adaptations pour les personnes atteintes de handicap moteur peuvent inclure des aides techniques telles que des fauteuils roulants, des béquilles, des orthèses ou des prothèses, ainsi que des adaptations de l'environnement de travail ou de la maison pour faciliter la mobilité et l'accessibilité.

1.1.4 Handicap mental



Le handicap mental est un type de handicap qui affecte la capacité mentale d'une personne, y compris sa capacité à penser, à apprendre, à se souvenir, à résoudre des problèmes et à communiquer. Ce handicap peut être causé par des lésions cérébrales, des troubles neurologiques, des troubles du développement ou des conditions génétiques. Il est généralement présent dès la naissance ou se développe pendant l'enfance.

Les personnes en situation de handicap mental peuvent rencontrer des difficultés à comprendre des informations complexes, à se rappeler des événements passés ou à traiter des informations nouvelles. Les adaptations pour les personnes atteintes de handicap mental peuvent inclure des aides pour la mémoire, des programmes de formation spécialisés, des environnements structurés et des adaptations de l'environnement de travail ou de la maison pour faciliter la communication et la compréhension.

Il est important de distinguer le handicap mental dû à une déficience intellectuelle, telle que la trisomie 21, du handicap cognitif qui affecte une ou plusieurs fonctions cognitives, comme les troubles dyslexiques ou le TDA(H).

1.1.5 Handicap psychique



Le handicap psychique est un type de handicap qui affecte la santé mentale d'une personne. Il se caractérise par des troubles psychiques ou des maladies mentales qui peuvent avoir un impact significatif sur la vie quotidienne et les capacités fonctionnelles d'une personne. Les handicaps psychiques peuvent inclure des troubles tels que la dépression, l'anxiété, la schizophrénie, les troubles bipolaires, les troubles de la personnalité, et bien d'autres. Les handicaps psychiques peuvent varier en termes de gravité, de durée et d'impact sur la vie de la personne.

Ils peuvent affecter divers aspects de la vie, tels que les relations sociales, l'emploi, les activités quotidiennes, la gestion des émotions et le bien-être général.

Il est essentiel de promouvoir une compréhension et une sensibilisation accrues envers les handicaps psychiques, de lutter contre la stigmatisation et de fournir un soutien adéquat aux personnes touchées. Cela peut inclure un accès à des soins de santé mentale, des thérapies appropriées, des programmes de réadaptation, des mesures d'adaptation en milieu de travail et un soutien social et émotionnel.

1.1.6 Handicap cognitif

Le handicap cognitif fait référence à un ensemble de difficultés ou de limitations qui affectent les fonctions cognitives d'une personne. Les fonctions cognitives comprennent des processus mentaux tels que la mémoire, l'attention, la perception, le raisonnement, la résolution de problèmes, la planification, l'organisation, le langage et les compétences exécutives. Le handicap cognitif peut également englober les troubles dyslexiques et le trouble du déficit de l'attention avec ou sans hyperactivité (TDA(H)), qui affectent certaines fonctions cognitives.

Pour aider les personnes atteintes d'un handicap cognitif à surmonter ces difficultés et à vivre de manière plus indépendante, diverses stratégies de compensation peuvent être mises en place. Les supports visuels, les routines structurées, les aménagements environnementaux, les aides à la mémoire, les techniques d'apprentissage adaptées et les soutiens social et accompagnement.

2 Handicap et accessibilité

Le handicap et l'accessibilité sont étroitement liés car l'accessibilité est un moyen essentiel de réduire les obstacles auxquels sont confrontées les personnes en situation de handicap. L'accessibilité permet de rendre l'environnement physique, social, culturel et technologique plus facilement utilisable pour tous et en particulier pour les personnes en situation de handicap. L'accessibilité permet ainsi de favoriser leur inclusion sociale et leur participation à la vie de la communauté. Elle peut également contribuer à améliorer la qualité de vie de ces personnes en leur offrant plus d'autonomie, d'indépendance et de possibilités. Elle contribue également à promouvoir l'inclusion sociale et la diversité, ainsi qu'à renforcer la cohésion sociale et le vivre-ensemble.

Le tableau ci-dessous présente une liste des divers types de handicap et leurs principaux défis lors de l'utilisation du web.

Tableau 01 : Principales difficultés rencontrées sur le web

Type de handicap	Principales difficultés sur le web
Visuel	Lecture des contenus en raison de problèmes de vision.
	Visualisation des images et des vidéos sans alternatives textuelles ou descriptions.
	Interaction avec des interfaces mal adaptées aux lecteurs d'écran.
Auditif	Accès aux contenus audio tels que vidéos et sons non sous-titrés.
	Absence de transcription ou de langage des signes pour les contenus audiovisuels.
	Difficultés à comprendre les informations uniquement basées sur l'audio.
Moteur	Utilisation difficile de la souris, du clavier ou des dispositifs d'entrée.
	Navigation et interaction avec des contenus interactifs non accessibles via le clavier.
	Accès limité aux fonctionnalités nécessitant des mouvements précis.
Cognitif	Compréhension des informations complexes et difficiles à suivre.
	Navigation et orientation dans des interfaces peu intuitives.
	Gestion de la surcharge d'informations et de distractions.
Psychique	Gestion de l'anxiété et du stress provoqués par certaines interactions en ligne.
	Besoin de contenus apaisants et d'options pour personnaliser l'expérience.
Mental	Traitement de l'information, concentration et orientation limités.
	Interaction avec des interfaces peu claires, nécessitant des efforts cognitifs supplémentaires.

La conception d'un site web accessible implique de prendre en compte ces défis et d'adopter des pratiques de conception inclusives pour garantir que tous les utilisateurs, quel que soit leur type de handicap, puissent accéder au contenu de manière efficace et sans obstacles. Les directives WCAG du W3C (Web Content Accessibility Guidelines) fournissent des recommandations détaillées pour créer des sites web accessibles.

2.1 Qu'est-ce que l'accessibilité numérique ?

L'accessibilité numérique est donc une extension de l'accessibilité, qui se concentre sur la création de sites web, de documents électroniques et d'applications mobiles qui peuvent être utilisés de manière efficace par tous. L'importance de l'accessibilité numérique a été reconnue par de nombreux gouvernements et organisations à travers le monde, qui ont adopté des réglementations et des normes pour garantir une accessibilité équitable aux personnes handicapées.

En France, la loi du 11 février 2005 pour l'égalité des droits et des chances, la participation et la citoyenneté des personnes handicapées a rendu obligatoire l'accessibilité des sites web publics, ainsi que des sites web privés proposant des services au public. L'accessibilité des sites web selon la loi de 2005 repose sur les dispositions réglementaires détaillées dans le Référentiel Général d'amélioration de l'accessibilité (RGAA). Ce référentiel fournit des critères techniques et des recommandations pour rendre les sites web accessibles.

Au niveau international, le World Wide Web Consortium (W3C) a développé des normes d'accessibilité web appelées Web Content Accessibility Guidelines (WCAG). Les WCAG sont régulièrement mises à jour pour tenir compte des avancées technologiques et des besoins des utilisateurs.

2.2 WCAG et RGAA : référentiels d'accessibilité pour les sites web

Ces deux référentiels d'accessibilité visent à rendre les sites web accessibles à un large éventail de handicaps, tels que les handicaps visuels, auditifs, moteurs, cognitifs, etc. Ils fournissent des critères techniques détaillés et des recommandations pour améliorer l'accessibilité des sites web, en tenant compte des technologies d'assistance utilisées par les personnes en situation de handicap, comme les lecteurs d'écran, les navigateurs vocaux, etc.

Les **WCAG** (Web Content Accessibility Guidelines) et le **RGAA** (Référentiel Général d'Amélioration de l'Accessibilité) sont tous deux des référentiels d'accessibilité pour les sites web, mais ils ont été développés pour des contextes différents.

2.2.1 Web Content Accessibility Guidelines (WCAG)

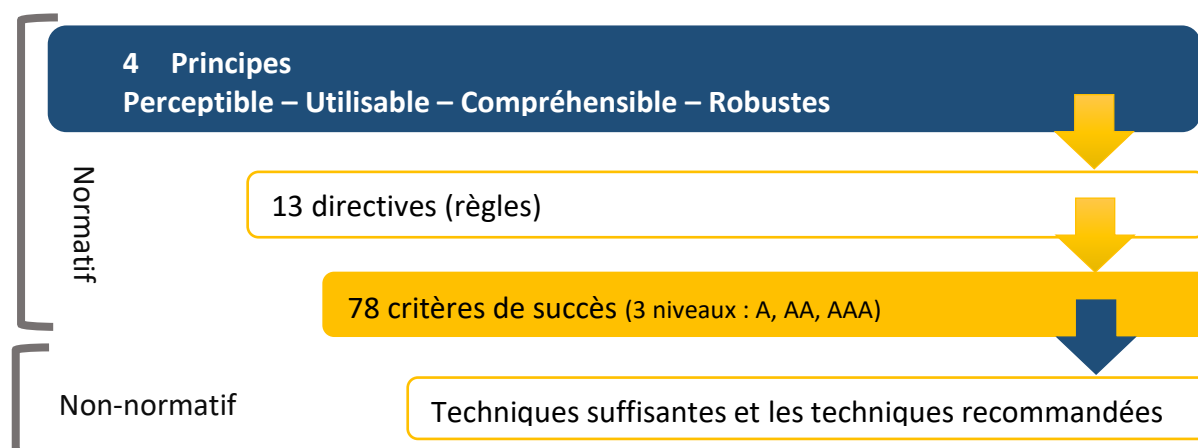
Les WCAG (Web Content Accessibility Guidelines) sont un ensemble de directives internationales développées par le World Wide Web Consortium (W3C) pour améliorer l'accessibilité des sites web pour les personnes en situation de handicap. Elles sont largement utilisées et reconnues à l'échelle mondiale. Les WCAG définissent trois niveaux de conformité : A, AA et AAA, qui indiquent le degré d'accessibilité atteint par un site web.

Les WCAG fournissent des directives détaillées pour rendre les contenus web accessibles à un large éventail de handicaps, tels que les handicaps visuels, auditifs, moteurs, cognitifs, etc. Elles couvrent différents aspects de l'accessibilité, y compris : La structure du contenu, la présentation visuelle, les médias et les alternatives, les formulaires et les fonctionnalités interactives et la compatibilité avec les technologies d'assistance

Les WCAG sont régulièrement mises à jour pour refléter les avancées technologiques et les évolutions des normes d'accessibilité. Il est important pour les concepteurs et les développeurs de sites web de se conformer aux WCAG afin d'améliorer l'accessibilité et de rendre leurs contenus disponibles pour tous les utilisateurs, quel que soit leur handicap.

La figure suivante donne une description générale de la structure des WCAG 2.1.

Figure 1 : La structure¹ des WCAG 2.1



Les WCAG 2.1 sont divisées en quatre principes² fondamentaux qui sont considérés comme les piliers de l'accessibilité web. Chaque principe est associé à des critères de succès spécifiques qui doivent être respectés pour atteindre un niveau de conformité donné.

❖ **Perceptible**

Les composants de l'information et de l'interface utilisateur doivent être présentables aux utilisateurs de manière à ce qu'ils puissent les percevoir. Cela signifie que les utilisateurs doivent être en mesure de percevoir l'information présentée, en utilisant des moyens adaptés tels que des descriptions alternatives pour les images, des sous-titres pour les vidéos, etc.

❖ **Utilisable**

Les composants de l'interface utilisateur et la navigation doivent être utilisables. Cela signifie que les utilisateurs doivent être en mesure d'utiliser l'interface (l'interface ne peut pas nécessiter une interaction qu'un utilisateur ne peut pas effectuer).

❖ **Compréhensible**

L'information et le fonctionnement de l'interface utilisateur doivent être compréhensibles. Cela signifie que les utilisateurs doivent être en mesure de comprendre les informations ainsi que le fonctionnement de l'interface utilisateur (le contenu ou l'opération ne peut pas être au-delà de leur compréhension).

❖ **Robuste**

Le contenu doit être suffisamment robuste pour pouvoir être interprété de manière fiable par une grande variété d'agents utilisateurs, y compris les technologies d'assistance. Cela signifie que les utilisateurs doivent être en mesure d'accéder au contenu au fur et à mesure que les technologies progressent (à mesure que les technologies et les agents utilisateurs évoluent, le contenu doit rester accessible).

Remarque : si l'une de ces conditions n'est pas respectée, les utilisateurs en situation de handicap ne pourront pas utiliser le Web.

¹ <https://www.w3.org/Translations/WCAG21-fr/>

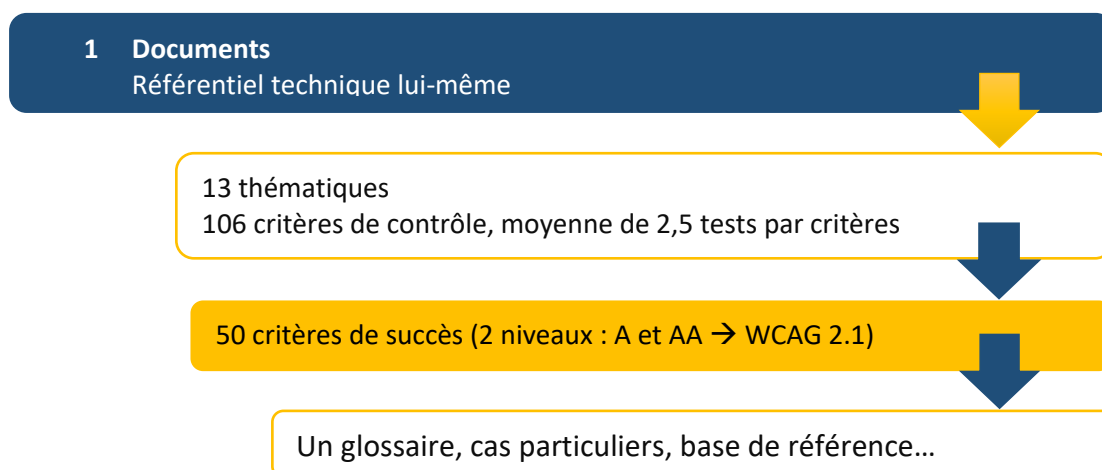
² <https://www.w3.org/WAI/WCAG21/Understanding/intro#understanding-the-four-principles-of-accessibility>

2.2.2 Référentiel d'amélioration de l'accessibilité (RGAA)

Le Référentiel Général d'Amélioration de l'Accessibilité (RGAA) est un référentiel spécifique à la France qui vise à garantir l'accessibilité des sites web des administrations publiques ou privé. Il a été développé par la Direction Interministérielle du Numérique (DINUM) et est régulièrement mis à jour pour suivre les évolutions technologiques et les normes d'accessibilité. Le RGAA est basé sur les WCAG (Web Content Accessibility Guidelines) en adaptant les critères d'accessibilité aux exigences légales et aux spécificités françaises. Il fournit des recommandations et des exigences supplémentaires pour assurer une accessibilité optimale des sites web en France.

Le RGAA est conçu pour répondre aux exigences de la législation française en matière d'accessibilité numérique, notamment la loi du 11 février 2005 pour l'égalité des droits et des chances des personnes en situation de handicap. Il propose une méthodologie pour évaluer la conformité d'un site web aux critères d'accessibilité. Il fournit des indications sur les tests à réaliser, les outils à utiliser et les rapports à générer pour évaluer le niveau d'accessibilité d'un site. Le RGAA définit deux niveaux de conformité similaire aux WCAG : A et AA. Ces niveaux indiquent le degré d'accessibilité atteint par un site web, avec AA étant recommandé pour les sites web en France. La version en vigueur du RGAA est la 4.1 et a été publiée le 18 février 2021³. La figure suivante donne une description générale de la structure du RGAA 4.1.

Figure 1 : La structure⁴ de RGAA 4.1



En résumé, bien que le WCAG et le RGAA soient basés sur des recommandations communes pour l'accessibilité des sites web, le WCAG est un référentiel international pour l'accessibilité des sites web, tandis que le RGAA est un référentiel national spécifique à la France qui s'appuie sur les normes WCAG pour s'assurer que les sites web français sont accessibles à tous.

2.2.3 Niveaux de conformité du web⁵

La conformité d'un site web est déterminée par son respect des normes d'accessibilité définies, telles que les WCAG 2.1 ou le RGAA 4.1, pour atteindre un niveau de conformité spécifique (niveau A, AA ou AAA).

³ <https://design.numerique.gouv.fr/accessibilite-numerique/rgaa/>

⁴ <https://accessibilite.numerique.gouv.fr/>

⁵ <https://www.w3.org/Translations/WCAG21-fr/#conformance>

Les niveaux de conformité du web selon les normes WCAG 2.1 et RGAA 4.1 sont les suivants :

✓ **Niveau A** (exigences minimales)

Le niveau A est le niveau le plus bas de conformité, qui comprend les exigences minimales en matière d'accessibilité web.

✓ **Niveau AA** (exigences intermédiaires)

Le niveau AA est le niveau intermédiaire de conformité, qui inclut des exigences plus strictes que le niveau A.

✓ **Niveau AAA** (exigences avancées)

Le niveau AAA est le niveau le plus élevé de conformité, qui inclut des exigences encore plus strictes que le niveau AA.

En général, le niveau AA est considéré comme le niveau de conformité souhaitable pour les sites web, car il offre un niveau élevé d'accessibilité tout en étant plus réaliste à atteindre que le niveau AAA, qui peut être très difficile à mettre en œuvre pour certains types de contenu.

2.3 Conformité et validation du web

La "conformité" et la "validation" dans le contexte du web font référence à la manière dont les sites web et les documents en ligne respectent les normes et les spécifications techniques établies. Cela contribue à assurer une expérience utilisateur cohérente, une accessibilité optimale et une interopérabilité entre les différentes plateformes et navigateurs.

2.3.1 Outils de conformité du web

Il existe plusieurs outils en ligne qui permettent d'évaluer automatiquement le niveau de conformité d'un site web aux normes d'accessibilité WCAG 2.1 et RGAA 4.1. Ils vérifient les pages web du site en question et fournissent un rapport détaillé sur les problèmes d'accessibilité identifiés. Le tableau 2 présente quelques outils pour vérifier le niveau de conformité du web.

Tableau 2 : Quelques outils pour vérifier le niveau de conformité

Outils	Descriptions	Liens
WAVE	Cet outil permet de vérifier les pages web en fonction des critères de conformité WCAG 2.1 et de repérer les erreurs d'accessibilité, notamment le contraste des couleurs, la présence d'alternatives textuelles pour les images, la facilité de navigation à l'aide du clavier, la lisibilité du texte.	Gratuit https://wave.webaim.org/
Axe DevTools	Axe DevTools est une extension pour Google Chrome et Mozilla Firefox qui permet d'auditer l'accessibilité des pages web en temps réel.	Gratuit https://www.deque.com/axe/devtools/
Achecker	Cet outil permet de vérifier la conformité aux normes d'accessibilité WCAG 2.1 et RGAA 4.1, ainsi que la convivialité et la lisibilité des pages web. Il fournit des recommandations pour améliorer l'accessibilité du site.	Gratuit https://achecker.acks.ca/checker/
Pa11y	Pa11y est un outil open-source qui permet de vérifier la conformité d'une page web. Il fournit des rapports détaillés pour aider les développeurs à corriger les erreurs d'accessibilité.	Gratuit https://pa11y.org/

Remarque : ces outils peuvent aider les développeurs et les designers à vérifier la conformité de leurs sites web avec les normes d'accessibilité, ce qui peut aider à améliorer l'accessibilité et l'expérience utilisateur pour tous.

2.3.2 Outils de contraste sur le web

Ce sont des outils qui permettent de vérifier la conformité des couleurs utilisées sur un site web avec les normes d'accessibilité en vigueur. Le contraste est un élément important en matière d'accessibilité, car il permet aux personnes atteintes de troubles de la vision de distinguer facilement les différents éléments de la page. Le tableau 3 présente quelques outils pour vérifier les contrastes sur le web.

Tableau 3 : Quelques outils pour vérifier les contrastes sur le web

Outils	Descriptions	Liens
Accessibility Insights for Web	Cette extension de navigateur permet de détecter les problèmes d'accessibilité sur une page web, y compris le contraste des couleurs. Il fournit des recommandations pour corriger les problèmes détectés.	Gratuit https://accessibilityinsights.io/
A11y Color Contrast Accessibility Validator	Cet outil permet de vérifier le contraste des couleurs selon les normes WCAG 2.1. Il propose également des conseils pour améliorer l'accessibilité de votre site web.	Gratuit https://color.a11y.com/
Color Contrast Analyzer	Cet outil développé par Paciello Group permet de vérifier le contraste des couleurs et de fournir des recommandations en fonction des critères WCAG et RGAA. Il propose des simulations pour différents types de déficiences visuelles.	Gratuit https://www.paciellogroup.com/resources/contrastanalyzer/
Colorable	Cet outil permet de vérifier le contraste entre les couleurs utilisées sur une page web, en particulier pour les éléments de texte et de fond. Il fournit des recommandations pour choisir des couleurs avec un contraste élevé pour améliorer l'accessibilité	Gratuit https://colorable.jxnblk.com/
Contrast Checker de WebAIM	Cet outil permet de vérifier le contraste des couleurs selon les normes WCAG 2.1 et le RGAA 4.1. Il suffit de saisir les codes hexadécimaux ou RGB des couleurs pour obtenir un rapport de contraste.	Gratuit https://webaim.org/resources/contrastchecker/
Stark	Stark est une extension de navigateur qui permet de vérifier les contrastes des couleurs directement sur une page web. Elle est disponible pour Chrome et Figma.	Gratuit/payant https://www.starkdesignsystem.com/
Tanaguru Contrast Finder	Cet outil permet de vérifier le contraste des couleurs selon les normes WCAG 2.1. Il propose des simulations pour différents types de déficiences visuelles.	Gratuit https://contrast-finder.tanaguru.com/
Tingtun Checker	Cet outil permet de vérifier le contraste des couleurs sur un site web. Il suffit d'entrer l'URL du site à vérifier et l'outil affiche une liste des erreurs de contraste ainsi que des suggestions pour les corriger.	Gratuit https://www.tingtun.no/check/

Remarque : en utilisant ces outils, les designers, les développeurs et autres peuvent s’assurer que les couleurs utilisées sur leur page web sont accessibles à tous les utilisateurs.

2.3.3 Outils de validation

Ils permettent de vérifier si le code HTML, CSS et JavaScript du site web est correct et conforme aux normes. Ces outils de validation vous aident à identifier les erreurs, les problèmes de syntaxe, les incohérences ou les pratiques déconseillées dans votre code. Le tableau 4 présente quelques outils qui permet de maintenir un code de qualité tout au long du processus de développement.

Tableau 4 : Quelques outils pour vérifier la validation du code

Outils	Descriptions	Liens
CSS Validator	Un outil de validation en ligne pour vérifier la conformité du code CSS aux spécifications du W3C.	Gratuit https://jigsaw.w3.org/css-validator/
ESLint	Un outil de validation pour le code JavaScript avec de nombreuses règles configurables pour détecter les erreurs courantes et les pratiques de codage incohérentes.	Gratuit https://eslint.org/
HTML Validator	Un outil de validation en ligne pour vérifier la conformité du code HTML aux normes du W3C.	Gratuit https://validator.w3.org/
JSHint	Outil pour vérifier la validité du code JavaScript. Il vérifie le code JavaScript pour les erreurs courantes et les problèmes de style.	Gratuit https://jshint.com/
Total Validator	Un outil de validation de site web qui vérifie la conformité aux normes d’accessibilité WCAG 2.1 et RGAA 4.1, ainsi que la validité du code HTML et CSS.	Gratuit/Payant https://www.totalvalidator.com/

Remarque : Ces outils peuvent aider à détecter les erreurs de validation, mais ils ne garantissent pas une conformité totale aux standards du web. Une vérification manuelle par un expert en accessibilité numérique est souvent nécessaire pour s'assurer que tous les problèmes ont été résolus et que le site est pleinement conforme aux normes d'accessibilité.

2.3.4 Outils de tests manuels

Certains tests d’accessibilité ne peuvent pas être effectués automatiquement et nécessitent une vérification manuelle. Il existe des outils qui aident les testeurs à effectuer ces tests manuellement et à suivre un processus de test structuré. Le tableau 5 présente quelques outils de tests manuels pour vérifier l’accessibilité du web.

Tableau 5 : Quelques outils de tests manuels pour vérifier l’accessibilité du web.

Outils	Descriptions	Liens
Lecteur d’écran	NVDA, JAWS, ou VoiceOver peuvent être utilisés pour tester l’accessibilité des sites web pour les utilisateurs malvoyants ou non-voyants.	Gratuit https://www.nvaccess.org/
Testeurs clavier	KeyboardTester peut être utilisé pour tester l’accessibilité des sites web pour les utilisateurs qui ne peuvent pas utiliser la souris.	Gratuit https://www.paciellogroup.com/resources/wat/
Tota11y	L’outil fournit une superposition (overlay) sur la page web qui met en évidence les éléments de la page qui peuvent être améliorés en termes d’accessibilité, tels que les éléments sans texte alternatif ou les contrastes de couleurs insuffisants.	Gratuit https://khan.github.io/tota11y/

Remarque : Les outils de test automatisés et manuels sont tous deux importants pour garantir une accessibilité optimale. Les outils de test automatisés ne peuvent pas détecter tous les problèmes d'accessibilité, et il est donc essentiel de compléter ces tests avec des tests manuels.

2.4 Squelette de base pour une page web accessible

Un squelette de base pour une page web HTML accessible est une structure HTML qui est conçue pour être utilisée par tous. Il doit inclure des balises sémantiques pour aider les lecteurs d'écran à comprendre la structure de la page, des attributs d'accessibilité pour améliorer l'accessibilité des pages web et un contenu clair et facilement compréhensible. Le code 1 présente un squelette de base pour une page web accessible.

Code 1 : Les éléments clés d'un squelette de base pour une page web accessible

```
<DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="UTF-8">
    <title>Site de Pole Art et handicap</title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
  </head>
  <body>
    <header>
      <h1>Guide d'accessibilité numérique</h1>
      <nav>
        <ul>
          <li><a href="#">Accueil</a></li>
          <li><a href="#">Services</a></li>
          <li><a href="#">Contact</a></li>
          <li><a href="#">Accessibilité</a></li>
        </ul>
      </nav>
    </header>
    <main>
      <article>
        <h2>Accessibilité numérique</h2>
        <p>conception de produits et de services numériques qui sont accessibles
          et utilisables pour tous</p>
        <section>
          <h2>Règles de l'accessibilité </h2>
          <p>visent à garantir que les personnes en situation de handicap</p>
        </section>
      </article>
    </main>
    <footer>
      <p>Copyright © 2023</p>
    </footer>
  </body>
</html>
```

Explication du code :

La balise de titre « **title** » décrit le titre de la page. Les balises de titre « **h1, h2** » sont utilisées pour indiquer la hiérarchie des informations présentées sur la page. Les balises sémantiques « **header, main, article, section, footer** » sont utilisées pour diviser le contenu en sections logiques, et les balises appropriées « **nav, ul, li** » sont utilisées pour créer des éléments de navigation, des listes.

Tout cela contribue à améliorer la compréhension et à rendre les pages web plus accessibles et ergonomiques pour les utilisateurs en situations de handicap. En effet, l'utilisation de ces balises permet aux développeurs web et aux créateurs de contenu de fournir des informations claires et structurées sur la hiérarchie et le contenu de la page. Cela est particulièrement utile pour les lecteurs d'écran et autres technologies d'assistance, car cela leur permet de mieux comprendre le contenu de la page.

3 Les principes de base de l'accessibilité numérique

Les règles de l'accessibilité numérique visent à garantir que les personnes en situation de handicap, notamment celles ayant des déficiences visuelles, auditives, motrices ou cognitives, peuvent accéder aux contenus numériques de manière équitable. Voici quelques règles clés de l'accessibilité numérique :

3.1 Clarté

La clarté d'un site web est un concept qui renvoie à la facilité avec laquelle les utilisateurs peuvent comprendre l'information présentée sur le site. Quelques éléments pour améliorer la clarté d'un site web.

3.1.1 Simplicité du design

Un design simple et épuré signifie que l'interface du site est dépourvue d'éléments superflus et d'effets visuels trop complexes. Un design minimaliste permet aux utilisateurs de se concentrer sur l'information principale sans être distraits par des éléments inutiles. Des couleurs cohérentes et une mise en page équilibrée aident également à rendre le contenu plus lisible et compréhensible.

3.1.2 Qualité du contenu

Le contenu clair et concis est essentiel pour transmettre efficacement l'information aux utilisateurs. Évitez les discours complexes ou les phrases longues et privilégiez un langage simple et direct. Utilisez des paragraphes courts et des puces pour découper l'information en unités plus digestes. Une utilisation appropriée des titres et des sous-titres facilite la navigation dans le contenu et permet aux utilisateurs de repérer rapidement les parties qui les intéressent.

3.1.3 Structure et organisation

La structure du site doit être logique et cohérente. Les informations doivent être organisées de manière à ce que les utilisateurs puissent suivre un chemin intuitif à travers le site. Utilisez des catégories claires et des menus de navigation bien organisés pour permettre aux utilisateurs de trouver ce qu'ils cherchent sans effort. Veillez à ce que les liens internes soient pertinents et dirigent les utilisateurs vers des pages associées.

3.1.4 Accessibilité

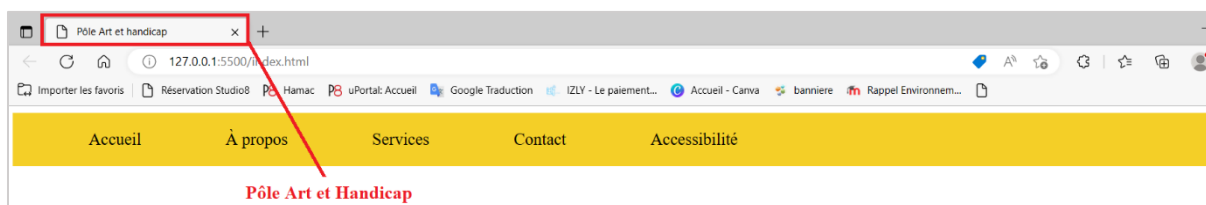
L'accessibilité est cruciale pour garantir que tous les utilisateurs, y compris ceux ayant des handicaps, puissent accéder au contenu du site. Respectez les normes d'accessibilité pour garantir que le site est utilisable par tous, quel que soit leur niveau de capacité. Fournissez des alternatives textuelles pour les images et les vidéos afin que les utilisateurs ayant des lecteurs d'écran puissent comprendre le contenu visuel.

Remarque : En intégrant ces éléments dans la conception de votre site web, vous créez une expérience utilisateur plus claire, conviviale et inclusive, ce qui améliorera la compréhension et la satisfaction des utilisateurs.

3.2 Balise de titre « title » pertinente

La balise de titre « **title** » est une balise HTML qui est utilisée pour définir le titre de la page. Elle apparaît dans la barre de titre du navigateur web et s'affiche également dans les résultats de recherche des moteurs de recherche. La figure suivante présente le titre du site.

Figure 1 : Affichage du titre du site web "Pole Art et Handicap".



Code 2 : exemple de code HTML pour un titre pertinent

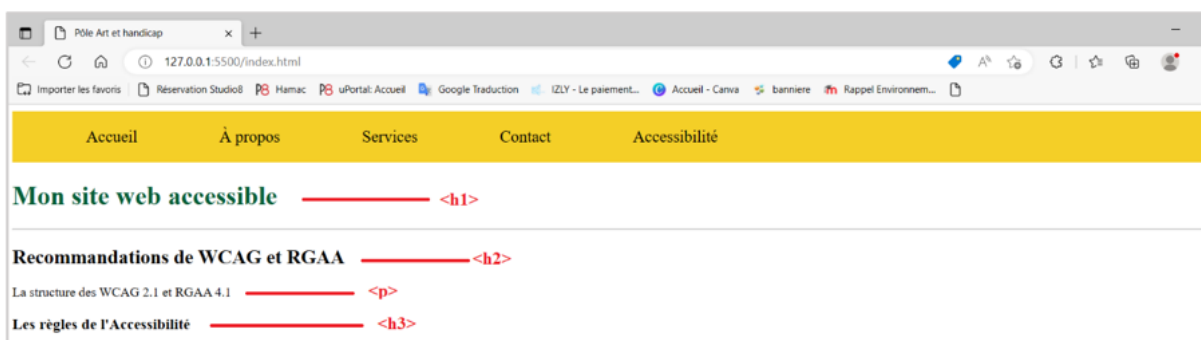
```
<title>Site de Pole Art et handicap</title>
```

L'utilisation d'une balise de titre « **title** » pertinente est importante pour l'accessibilité du web car elle permet aux utilisateurs d'avoir une idée claire du contenu de la page. Les usagers qui utilisent des technologies d'assistance telles que des lecteurs d'écran ou des navigateurs braille peuvent accéder à la balise de titre pour comprendre le contexte de la page.

3.3 Utiliser des balises de titres appropriées

Utiliser les balises de titre pour structurer le contenu d'une page et indiquer la hiérarchie des informations présentées. Les balises de titre vont de h1 à h6, le « h1 » est utilisé pour le titre principal de la page, qui devrait être unique sur chaque page, les h2 pour les titres de section, h3 pour les sous-titres de section et les h4 à h6 étant des sous-titres encore plus petits, si nécessaire. Évitez d'utiliser des balises de titre pour des éléments qui ne sont pas des titres ou des sous-titres.

Figure 2 : Utilisation adaptée des balises de titre



Code 3 : exemple de code HTML pour créer des balises appropriées

```
<h1>Mon site web accessible</h1>
<section id="section">
  <h2>Recommandations de WCAG et RGAA </h2>
  <p> La structure des WCAG 2.1 et RGAA 4.1 </p>
  <h3>Les règles de l'Accessibilité </h3>
</section>
```

En résumé, utilisez les balises de titre de manière adaptée, claire et cohérente pour structurer le contenu et améliorer la lisibilité du web. En utilisant correctement les balises de titre, vous pouvez rendre votre contenu plus facile à comprendre par les lecteurs et les moteurs de recherche.

3.4 Contenu alternatif

Fournir des alternatives textuelles et audio aux éléments visuels tels que les images, les vidéos et les graphiques.

❖ Image porteuse d'information

Les images peuvent être importantes pour communiquer des informations, mais elles ne sont pas accessibles aux personnes en situation de handicap visuel. Pour cette raison, les images doivent être accompagnées d'un texte alternatif décrivant leur contenu de manière claire et concise.



Pour créer une image porteuse d'information en HTML, vous pouvez ajouter un attribut « alt » à la balise « img ». Cet attribut fournit une description textuelle de l'image qui peut être utilisée par les lecteurs d'écran et les navigateurs pour personnes aveugles ou malvoyantes afin de comprendre le contenu de l'image.

Code 4 : exemple de code HTML pour insérer une image porteuse d'information

```

```

Une image porteuse d'information est une image qui a pour but de transmettre une information à l'utilisateur.

❖ Image de décoration

C'est une image qui ne transmet aucune information importante, elle est considérée comme



une « image de décoration ». Dans ce cas, l'attribut « alt » de l'image est vide, ce qui signifie que la description textuelle est inexistante. L'attribut est utilisé pour indiquer aux lecteurs d'écran que cette image est uniquement destinée à des fins de décoration et qu'elle ne doit pas être lue à haute voix pour les utilisateurs malvoyants ou aveugles.

Code 5 : exemple de code HTML pour insérer une image de décoration

```

```

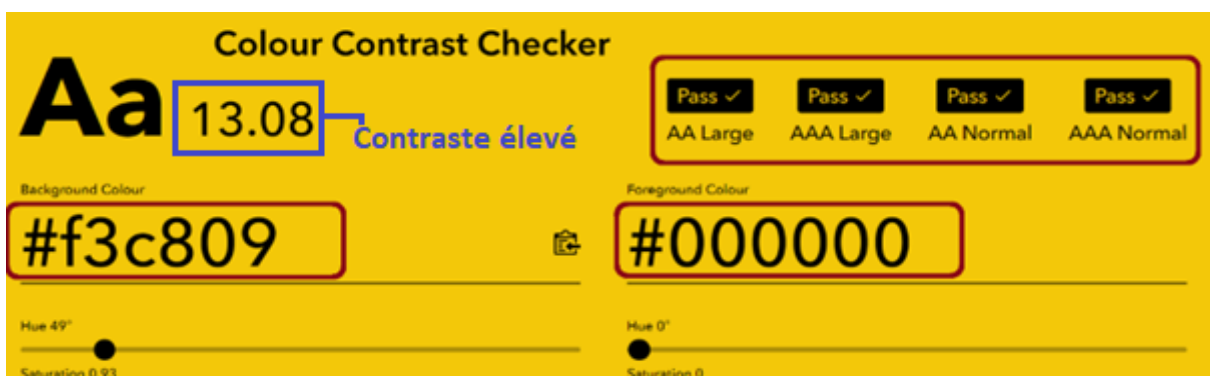
Aria-hidden= « true » est un attribut de l'accessibilité qui est utilisé pour indiquer aux lecteurs d'écran qu'un élément de la page web ne doit pas être lu à haute voix pour les utilisateurs en situation de handicap visuel.

3.5 Contraste élevé

Les personnes ayant des handicaps visuels peuvent rencontrer des problèmes pour distinguer les couleurs, ce qui souligne l'importance de rendre le texte et les images lisibles. Le contraste entre les couleurs doit être suffisant pour assurer une visibilité claire du contenu.

Pour garantir un contraste élevé, il est essentiel de choisir des couleurs nettement différentes, en particulier pour le texte et l'arrière-plan. Il existe des outils en ligne, comme le « [Coulour Contrast Checker](#) », qui permettent de vérifier le contraste entre les couleurs et de s'assurer de leur lisibilité (voir la figure suivante).

Figure 3 : Exemple de contraste élevé avec outils « Coulour Contrast Checker »

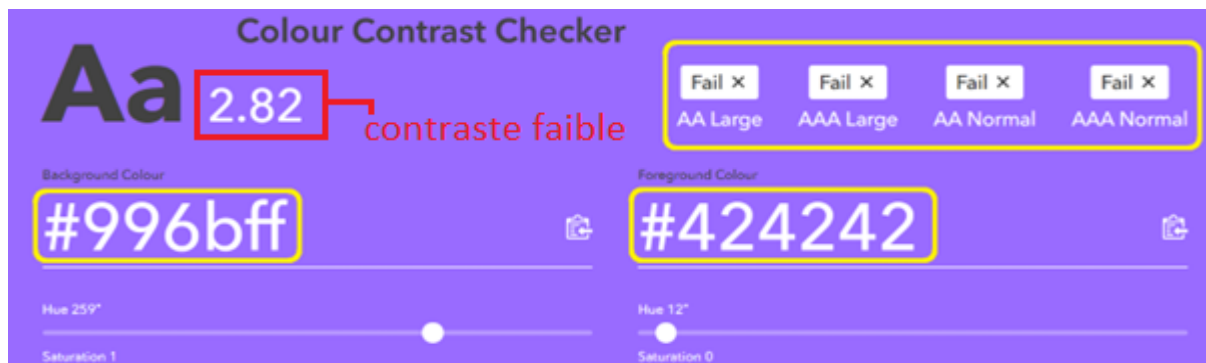


❖ Contraste faible

Un contraste faible sur le web peut nuire à l'accessibilité pour les personnes ayant une déficience visuelle, il est donc important de veiller à respecter les normes d'accessibilité pour offrir une expérience de navigation optimale à tous les utilisateurs.

Les normes WCAG 2.1 et RGAA 4.1 recommandent un ratio de contraste minimum de 4,5 :1 pour le texte de taille normale et de 3:1 pour le texte de grande taille et gras.

Figure 4 : Exemple de contraste trop faible



Colour Contrast Checker : est un outil en ligne qui permet de vérifier le contraste entre deux couleurs et de s'assurer qu'elles sont suffisamment contrastées pour une meilleure accessibilité web.

Remarque : Plusieurs outils en ligne pour vérifier le contraste entre les couleurs utilisées sur une page web, [référence tableau 3](#).

Quelques bonnes pratiques à suivre :

- Utilisez des couleurs contrastées pour le texte et le fond.

Exemple : **fond sombre → couleurs claires pour le texte et vice versa.**

- Évitez d'utiliser des couleurs similaires pour le texte et le fond.

Exemple : texte gris clair → fond blanc peut être difficile à lire pour certaines personnes.

- Utilisez des couleurs vives pour les éléments interactifs.

Exemple : **boutons** et les **liens**. Cela les rendra plus facilement identifiables et cliquables.

- Évitez les images de fond complexes qui peuvent rendre le texte difficile à lire.

Exemple : image de fond → couleur de texte qui contraste fortement avec l'image.

- Utilisez des polices de caractères simples et faciles à lire.

Exemple : Polices sans serif → plus faciles à lire : Arial, Helvetica, Calibri, Verdana

Polices serif → plus difficiles à lire : Times New Roman, Georgia, Cambria, Palatino

- Évitez d'utiliser des effets de texte.

Exemple : les ombres portées, qui peuvent rendre le texte flou et difficile à lire.

- Utilisez des outils en ligne pour vérifier le contraste entre les couleurs

Exemple : [référence tableau 3](#).

En suivant ces bonnes pratiques, vous pouvez créer une page web avec un contraste élevé qui est accessible à tous.

3.6 Navigation simple

La navigation d'un site web est l'ensemble des éléments qui permettent aux utilisateurs de se déplacer à travers les différentes pages et sections du site. Les éléments clés de la navigation comprennent :

3.6.1 Le menu principal

Un ensemble d'options de navigation qui permettent aux utilisateurs d'accéder aux pages principales du site.

Code 6 : exemple de code HTML pour créer un menu de navigation accessible

```
<nav role="navigation">
  <ul>
    <li><a href="#">Accueil</a></li>
    <li><a href="#">À propos</a></li>
    <li><a href="#">Services</a></li>
    <li><a href="#">Contact</a></li>
    <li><a href="#">Accessibilité</a></li>
  </ul>
</nav>
```

Explication du code :

La balise « **nav** » est utilisée pour indiquer qu'il s'agit d'un menu de navigation. Les éléments de menu sont représentés par des balises « **li** » imbriquées dans une balise « **ul** ». Chaque élément de menu est représenté par une balise « **a** » avec un attribut « **href** » correspondant à la page ou à la section correspondante du site.

3.6.2 Les liens dans une page web

Les liens sont des éléments interactifs d'une page Web qui permettent aux utilisateurs de naviguer vers d'autres pages web, documents ou ressources. Le rôle principal des liens est de connecter les différentes parties d'un site Web entre elles et de fournir une navigation intuitive pour les utilisateurs. Ils permettent aux utilisateurs de passer facilement d'une page à l'autre, de trouver des informations complémentaires et de découvrir de nouveaux contenus.

✓ Liens hypertextes :

Les liens hypertextes sont les liens les plus courants dans une page web. Ils permettent de relier une page à une autre page ou à une ressource externe. Pour un lien hypertexte accessible, il est important de fournir un texte de lien clair et descriptif, qui indique clairement le contenu de la page vers laquelle le lien pointe.

Code 7 : exemple de code HTML pour créer un lien hypertexte

```
<a href="https://www.imagolereseau.fr/" title="Accéder à la page d'accueil d'imagolereseau">imagolereseau</a>
```

Explication du code :

« **a** » est la balise pour créer un lien hypertexte en HTML, « **href** » est l'attribut qui spécifie l'URL de la page vers laquelle le lien doit pointer, dans ce cas-ci la page d'accueil d'Imago le réseau. « **title** » est l'attribut qui spécifie une description textuelle du lien pour les utilisateurs qui peuvent avoir des difficultés à interpréter l'icône ou le texte du lien. Dans ce cas-ci, il s'agit d'une courte description de la destination du lien, c'est-à-dire la page d'accueil d'Imago le réseau, « **imagolereseau** » est le texte du lien qui sera affiché à l'écran.

✓ Liens d'ancrage :

Ils permettent de lier une partie spécifique d'une page à une autre partie de la même page. Ils sont souvent utilisés pour créer des liens de navigation internes. Pour un lien d'ancrage accessible, il est important de fournir un texte de lien clair et descriptif, qui indique clairement la destination du lien.

Code 8 : exemple de code HTML pour créer un lien d'ancrage

```
<a href="#qui-sommes-nous" title="Aller à la section Qui Sommes-Nous"> Voir qui sommes-nous</a>
```

Explication du code :

« **a** » : c'est la balise HTML pour créer un lien hypertexte, **href="#qui-sommes-nous"** : c'est l'attribut qui indique l'URL de destination du lien. Ici, l'URL est un identifiant qui correspond à l'ancre de la section « Qui Sommes-Nous » sur la même page. L'attribut **title = "Aller à la section Qui Sommes-Nous"** : c'est l'attribut qui fournit une description textuelle supplémentaire pour aider les utilisateurs à comprendre la destination du lien. Ici, il indique que le lien mène à la section « Qui Sommes-Nous ». Voir qui sommes-nous : c'est le texte visible du lien qui sera affiché sur la page.

✓ Les Liens de courrier électronique :

Ils permettent aux utilisateurs de cliquer sur un lien pour envoyer un e-mail à une adresse e-mail spécifique. Pour un lien de courrier électronique accessible, il est important de fournir une adresse e-mail claire et descriptif, qui indique clairement le destinataire de l'e-mail.

Code 9 : exemple de code HTML pour créer un lien de courrier électronique

```
<a href="mailto:imagolereseau@gmail.com" aria-label="Envoyer un courrier électronique à Imago Le Réseau"> Envoyer un courrier électronique à Imago </a>
```

Explication du code :

«**a**» : Balise pour créer un lien hypertexte, **href** : attribut pour spécifier l'adresse de destination du lien. **mailto** : protocole spécifique pour les liens de courrier électronique. imagolereseau@gmail.com : adresse de courrier électronique de destination et **aria-label** : attribut pour ajouter une étiquette personnalisée qui décrit la fonction du lien, pour les utilisateurs de lecteurs d'écran. Envoyer un courrier électronique à Imago : Texte d'ancrage

du lien. L'exemple de code ci-dessus crée un lien de courrier électronique qui indique clairement sa fonction et fournit une alternative textuelle pour les utilisateurs de lecteurs d'écran.

Il est important de s'assurer que les liens d'un site web sont clairement identifiés, facilement accessibles et que leur destination est clairement indiquée pour garantir une expérience utilisateur optimale.

3.6.3 Les boutons dans une page web

Ils sont des éléments interactifs qui permettent à l'utilisateur de déclencher une action ou un changement sur la page. Ils sont souvent utilisés pour soumettre un formulaire, confirmer une action, afficher ou masquer du contenu, etc. Les boutons doivent être clairement identifiables visuellement et leur libellé doit être facilement compréhensible.

✓ Boutons d'action :

Ils sont utilisés pour déclencher une action spécifique, telle que l'envoi d'un formulaire ou la suppression d'un élément. Ils sont souvent de couleur vive et placés de manière à être facilement identifiables par les utilisateurs.

Code 10 : exemple de code HTML pour créer un bouton d'action

```
<button type="button" id="submit-form" aria-label="Envoyer le formulaire">Envoyer</button>
```

Explication du code :

«**button**» : la balise HTML `button` est utilisée pour créer le bouton, **type= " button "** : l'attribut `type` est défini à « **button** » pour indiquer que ce bouton n'a pas d'action spécifique. **id=" submit-form "** : l'attribut `id` est utilisé pour identifier ce bouton dans le code HTML.

aria-label= " Envoyer le formulaire " : l'attribut `aria-label` fournit une alternative textuelle pour les utilisateurs de lecteurs d'écran et permet aux utilisateurs de clavier de naviguer facilement jusqu'au bouton. Dans ce cas, il indique que le bouton permet d'envoyer un formulaire. Le texte « Envoyer » est inclus entre les balises de début et de fin du bouton pour indiquer aux utilisateurs de l'interface visuelle ce que fait le bouton.

✓ Boutons de navigation :

Ils sont utilisés pour permettre aux utilisateurs de naviguer entre différentes pages d'un site web. Ils peuvent être placés dans un menu de navigation ou directement sur la page.

Code 11 : exemple de code HTML pour créer un bouton de navigation

```
<button type="button" id="btn-navigation" title="Aller à la navigation"
  aria-label="Menu de navigation"> <a href="#section-nav">Navigation</a>
</button>
```

Explication du code :

La balise « **button** » est utilisée pour créer un bouton d'interaction. L'attribut **type= " button "** est utilisé pour spécifier que ce bouton est un bouton d'interaction. L'attribut **id="btn-navigation"** est utilisé pour donner un identifiant unique à ce bouton. L'attribut **title= " Aller à la navigation "** est utilisé pour fournir une description textuelle de la fonctionnalité du bouton. Cette description est affichée lorsqu'un utilisateur survole le bouton avec sa souris. L'attribut **aria-label= " Menu de navigation "** est utilisé pour fournir une alternative textuelle à l'intérieur du bouton pour les utilisateurs qui utilisent des technologies d'assistance, les lecteurs d'écran.

✓ Boutons de partage :

Ils permettent aux utilisateurs de partager le contenu de la page sur les réseaux sociaux ou par email. Ils sont souvent représentés par des icônes et placés en bas de la page ou à côté du contenu à partager.

Code 12 : exemple de code HTML pour créer un bouton de partage

```
<button type="button" id="btn-share" aria-label="Partager cette page">
  <a href="#" title="Partager sur les réseaux sociaux">Partager</a>
</button>
```

Explication du code :

La balise « **button** » est utilisée pour créer le bouton de partage. L'attribut **type= " button "** indique que le bouton est un bouton d'action. L'attribut **id = " btn-share "** est utilisé pour identifier le bouton. L'attribut **aria-label= " Partager cette page "** fournit une description du bouton pour les lecteurs d'écran. La balise « **a** » est utilisée pour créer un lien qui permettra d'accéder à une page pour partager la page courante sur les réseaux sociaux. L'attribut **title= " Partager sur les réseaux sociaux "** fournit une information supplémentaire sur l'action effectuée par le bouton de partage.

Remarque : Il est important que les boutons soient codés de manière accessible pour permettre une navigation facile aux personnes ayant des troubles de la vision ou des difficultés à utiliser une souris.

3.6.4 La barre de recherche

Elle permet aux utilisateurs de rechercher des informations spécifiques sur le site. Il s'agit généralement d'un champ de texte accompagné d'un bouton de soumission permettant à l'utilisateur de lancer sa recherche.

Code 13 : exemple de code HTML pour créer une barre de recherche

```
<form action="/search" method="get">
  <label for="search">Rechercher </label>
  <input type="text" id="search" name="nom de variable" aria-label="Rechercher sur le
    site" placeholder="Rechercher...">
  <button type="submit" aria-label="Lancer la recherche">Rechercher</button>
</form>
```


Explication du code :

La balise « **form** » définit un formulaire qui sera utilisé pour la recherche. L'attribut **action** indique l'URL à laquelle le formulaire sera envoyé une fois soumis.

L'attribut **method** spécifie la méthode HTTP utilisée pour envoyer les données du formulaire. La balise « **label** » est associée à l'élément `input` à l'aide de l'attribut **for** pour donner une description à la barre de recherche. L'attribut **id** de l'élément `input` est utilisé pour associer le label correspondant. L'attribut **name** de l'élément `input` spécifie le nom de la variable envoyée lors de la soumission du formulaire. L'attribut **aria-label** fournit une étiquette pour l'élément `input` qui sera lue par les technologies d'assistance. L'attribut **placeholder** donne un exemple de ce que l'utilisateur doit saisir dans la barre de recherche. Le bouton de soumission est défini à l'aide de la balise « **button** » avec l'attribut `type` spécifié comme `submit`.

3.7 Les filtres et les options de tri

Les filtres et les options de tri sont des fonctionnalités courantes dans les interfaces utilisateur. En termes de conception, il est important de présenter ces fonctionnalités de manière claire et accessible.

✓ Les filtres

Les filtres permettent aux utilisateurs de restreindre les résultats de recherche en fonction de certains critères, tels que la catégorie, la taille, la couleur, le prix, etc.

Les filtres doivent être placés de manière visible et facilement accessibles, de préférence dans une colonne latérale ou en haut de la page.

Code 14 : exemple de code HTML pour créer un filtre

```
<label for="filter">Filtrer par : </label>
<select id="filter" name="filter" aria-label="Filtrer par">
  <option value="option1" title="Tous les produits">Tous les produits</option>
  <option value="option2" title="Produits en stock">Produits en stock</option>
  <option value="option3" title="Produits en promotion">Produits en promo</option>
</select>
```

Explication du code :

La balise « **label** » est utilisée pour associer l'étiquette « Filtrer par : » à l'élément `select` en utilisant l'attribut `for`. L'attribut **aria-label** fournit une alternative textuelle pour les lecteurs d'écran. Chaque option est également accompagnée d'une description en utilisant l'attribut **title** pour aider les utilisateurs à comprendre les options disponibles.

✓ Les options de tri

Les options de tri permettent aux utilisateurs de trier les résultats selon certains critères, tels que le prix croissant/décroissant, la popularité, la date d'ajout, etc. Les options de tri doivent être facilement accessibles, généralement sous forme de liste déroulante ou de boutons.

Code 15 : exemple de code HTML pour créer une option de tri

```
<label for="tri">Trier par :</label>
<select id="tri" name="tri" aria-label="Options de tri">
  <option value="date">Date</option>
  <option value="nom">Nom</option>
  <option value="prix">Prix</option>
</select>
```

Explication du code :

La balise « **label** » permet d'associer un libellé à un champ de formulaire pour améliorer l'accessibilité. L'attribut **for** de la balise label fait référence à l'**id** du champ de formulaire associé. La balise « **select** » permet de créer une liste déroulante d'options. L'attribut **aria-label** est utilisé pour fournir une description textuelle des options de tri pour les utilisateurs de lecteurs d'écran. L'attribut « **value** » de chaque option représente la valeur qui sera envoyée lors de la soumission du formulaire.

Remarque : En ce qui concerne l'accessibilité, il est important de fournir des descriptions claires pour chaque filtre et option de tri, et de permettre l'utilisation des touches du clavier pour naviguer entre les options.

3.8 Commandes de clavier

La prise de focus est un élément clé pour améliorer l'accessibilité des pages web. Elle permet à l'utilisateur de savoir où il se trouve sur la page et de pouvoir interagir avec les différents éléments en utilisant uniquement le clavier. Ainsi, il est essentiel de rendre tous les éléments interactifs, tels que les liens, les éléments de formulaires et les boutons, accessibles via la touche de tabulation. La possibilité de naviguer facilement avec le clavier est particulièrement importante pour les personnes ayant des limitations physiques ou qui préfèrent utiliser un clavier plutôt qu'une souris pour naviguer sur le web. Cela permet une expérience de navigation plus fluide et plus efficace pour ces utilisateurs.

La prise de focus est cruciale pour les personnes malvoyantes ou aveugles utilisant un lecteur d'écran. Lorsqu'un élément est en focus, le lecteur d'écran l'annonce, aidant l'utilisateur à se repérer sur la page. Cela facilite la navigation et l'interaction en fournissant des informations sonores sur les éléments sélectionnés.

Code 16 : exemple de code HTML comprenant une barre de navigation et un formulaire, conçu pour faciliter l'interaction avec le clavier.

```
<nav>
  <ul>
    <li><a href="#">Accueil</a></li>
    <li><a href="#">À propos</a></li>
    <li><a href="#">Contact</a></li>
  </ul>
</nav>
<form>
  <label for="nom">Nom :</label>
  <input type="text" id="nom" name="nom">
  <label for="email">Email :</label>
  <input type="email" id="email" name="email">
  <button type="submit">Envoyer</button>
</form>
```

Explication du code :

Les liens sont représentés par des balises « a », les éléments de formulaires sont représentés par des balises « input » et « label », et le bouton est représenté par une balise « button ». L'attribut « for » est utilisé pour lier chaque **label** à son élément de formulaire correspondant.

En résumé, ce code crée une barre de navigation avec des liens vers différentes pages et un formulaire de saisie avec des champs de nom et d'e-mail, ainsi qu'un bouton pour soumettre le formulaire.

Code 17 : exemple de code CSS pour appliquer un focus

```
:focus {
  outline: 2px solid #000000;
  outline-offset: 2px;
}
```

Explication du code :

L'état : **focus** est utilisé pour appliquer une bordure autour de l'élément qui a le focus, afin de le mettre en évidence. Dans cet exemple, une bordure de 2 pixels est appliquée avec un décalage de 2 pixels pour éviter de couvrir le contenu de l'élément.

Code 18 : exemple de code HTML pour définir l'ordre de tabulation

```
<input type="text" tabindex="1">
<button tabindex="2">Rechercher</button>
```

Explication du code :

Ce code HTML concerne deux éléments : un champ de saisie de texte « input » et un bouton « button ». Les attributs **tabindex** sont utilisés pour définir l'ordre de navigation au clavier sur les éléments de la page. Cet élément « input » crée un champ de saisie de texte de type "text". L'attribut **tabindex** est défini à "1", ce qui signifie que cet élément aura la priorité la plus élevée lors de la navigation au clavier. Cet élément « button » crée un bouton avec le texte "Rechercher". L'attribut **tabindex** est défini à "2", ce qui indique que cet élément sera le deuxième à recevoir le focus lors de la navigation au clavier.

L'attribut « **tabindex** » est utilisé pour personnaliser l'ordre de navigation au clavier sur une page. Cela est utile pour améliorer l'accessibilité et l'expérience utilisateur, en permettant aux utilisateurs de naviguer efficacement sur la page en utilisant uniquement le clavier.

Quelques commandes de clavier courantes pour la navigation web :

Tab : permet de passer d'un élément interactif à un autre (les liens, les boutons et des champs de formulaire) ;

Shift + Tab : permet de passer d'un élément interactif à un autre dans l'ordre inverse ;

Flèches directionnelles : permettent de naviguer dans les menus déroulants et les listes ;

Entrée ou Espace : permettent d'activer un lien ou un bouton

Ctrl + F : permet de rechercher du contenu sur une page web

Ctrl + T : permet d'ouvrir un nouvel onglet dans le navigateur.

Ctrl + W : permet de fermer l'onglet actif dans le navigateur.

Ctrl + P : permet d'imprimer la page active dans le navigateur.

3.9 Sous-titres

Les sous-titres permettent aux personnes sourdes ou malentendantes de suivre le contenu de la vidéo en lisant les mots qui s'affichent à l'écran.

Code 19 : exemple de code HTML pour ajouter une vidéo avec des sous-titres

```
<video controls>
  <source src="ma_video.mp4" type="video/mp4">
  <track kind="subtitles" src="mes_sous-titres.vtt" srclang="fr" label="Français">
</video>
```

Explication du code :

La balise « **video** » est utilisée pour insérer une vidéo dans la page HTML. L'attribut « **controls** » ajoute des contrôles de lecture standard à la vidéo, tels que le bouton de lecture/pause, la barre de progression et les boutons de volume. La balise « **source** » est utilisée pour spécifier la source de la vidéo. L'attribut « **src** » indique l'emplacement du fichier vidéo "ma_video.mp4", tandis que l'attribut « **type** » indique le type MIME de la vidéo (ici, "video/mp4" pour un fichier MP4). La balise « **track** » est utilisée pour spécifier les sous-titres de la vidéo. L'attribut « **kind** » est défini sur "subtitles" pour indiquer qu'il s'agit de sous-titres.

L'attribut « **src** » indique l'emplacement du fichier de sous-titres "mes_sous-titres.vtt", tandis que l'attribut « **srclang** » spécifie la langue des sous-titres (ici, le français). L'attribut « **label** » fournit une étiquette descriptive pour les sous-titres en français.

Remarque : le format de fichier de sous-titres utilisé dans ce code est le WebVTT (.vtt). Le fichier de sous-titres doit être synchronisé avec la vidéo pour que les sous-titres s'affichent au bon moment.

3.10 Description audio

La description audio est le seul moyen de comprendre les informations visuelles des vidéos pour les personnes aveugles et malvoyantes qui ne peuvent pas voir les images. Par conséquent, il est essentiel de fournir une description complète et détaillée de toutes les informations visuelles importantes.

Code 20 : exemple de code HTML pour créer une description audio

```
<video controls>
  <source src="ma_video.mp4" type="video/mp4">
  <track kind="description" src="ma_description.mp3" />
</video>
```

Explication du code :

La balise « **track** » permet de spécifier la source de la description audio. L'attribut **kind** indique que la piste est du type « description », l'attribut **src** indique l'emplacement du fichier de description audio.

3.11 Taille de police réglable

Pour permettre aux utilisateurs d'ajuster la taille du texte pour répondre à leurs besoins visuels, on peut utiliser des styles de texte CSS qui permettent aux utilisateurs d'agrandir ou de réduire la taille du texte.

Code 21 : exemple de code CSS pour agrandir ou réduire la taille du texte.

```
body {
  font-size: 16px;
}
@media only screen and (max-width: 768px) {
  body {
    font-size: 12px;
  }
}
@media only screen and (max-width: 480px) {
  body {
    font-size: 8px;
  }
}
@media only screen and (max-width: 320px) {
  body {
    font-size: 6px;
  }
}
```

Explication du code :

Le style de base pour le texte est de 16 pixels (px). Cependant, en utilisant les règles CSS **@media**, on peut spécifier différentes tailles de texte pour différentes tailles d'écran.

On peut inclure des boutons ou des liens sur la page pour permettre aux utilisateurs d'augmenter ou de diminuer la taille du texte en cliquant sur un bouton plutôt que de zoomer la page.

Code 22 : exemple de code HTML pour ajouter des boutons.

```
<p>
  Vous pouvez inclure des boutons ou des liens sur votre page pour permettre aux
  utilisateurs d'augmenter ou de diminuer la taille du texte en cliquant sur un
  bouton plutôt que de zoomer la page.
</p>
<button onclick="increaseFontSize()">Agrandir</button>
<button onclick="decreaseFontSize()">Réduire</button>
```

Explication du code :

On a ajouté deux boutons qui permettent d'augmenter et de diminuer la taille du texte en cliquant sur eux. Lorsque l'utilisateur clique sur l'un de ces boutons, la fonction correspondante est appelée en JavaScript pour modifier la taille de police du texte.

Code 23 : exemple de code javascript pour agrandir ou réduire la taille du texte.

```
function increaseFontSize() {
  document.body.style.fontSize = "larger";
}
function decreaseFontSize() {
  document.body.style.fontSize = "smaller";
}
```

Explication du code :

Dans ce code, les fonctions **increaseFontSize()** et **decreaseFontSize()** sont définies pour modifier la taille de police du corps de la page. La fonction **increaseFontSize()** applique la propriété CSS « larger », ce qui agrandit la taille de police, tandis que la fonction **decreaseFontSize()** applique la propriété CSS « smaller », ce qui réduit la taille de police.

Code 24 : exemple de code CSS pour agrandir ou réduire la taille du texte

```
button {
  font-size: 16px;
  padding: 8px;
  background-color: #ffffff;
  border: 1px solid #000000;
  border-radius: 4px;
  cursor: pointer;
  margin-right: 8px;
}
button:hover {
  background-color: #000000;
  color: #ffffff;
}
```

Explication du code :

Les boutons sont affichés avec une police de 16 pixels, un fond blanc et une bordure de 1 pixel. Ils ont un rayon de bordure de 4 pixels, un curseur de souris de type pointeur et un espacement de 8 pixels entre eux. Lorsque l'utilisateur survole un bouton, le fond devient noir et la couleur de police devient blanche pour indiquer que le bouton est cliquable.

3.12 Choix de la langue

Pour assurer l'accessibilité du contenu de votre site web à des utilisateurs de différentes langues, il est important de permettre aux utilisateurs de choisir la langue de leur choix.

Code 25 : exemple de code HTML, qui permet aux utilisateurs de choisir la langue de leur choix

```
<select id="langSelect" onchange="changeLanguage()">
  <option value="fr">Français</option>
  <option value="en">English</option>
  <option value="es">Español</option>
</select>
```

Explication du code :

On a ajouté un sélecteur de langue avec des options pour le français, l'anglais et l'espagnol. On a également ajouté un attribut "onchange" qui appelle une fonction « **changeLanguage()** » chaque fois que l'utilisateur sélectionne une langue différente.

3.13 Tableaux et libellés

Les tableaux ont un rôle important en matière d'accessibilité sur le web car ils permettent de présenter des données structurées de manière claire et organisée.

Code 26 : exemple de code HTML, pour rendre les tableaux accessibles

```
<table>
  <caption>Liste des étudiants</caption>
  <thead>
    <tr>
      <th scope="col">Nom</th>
      <th scope="col">Prénom</th>
      <th scope="col">Âge</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th scope="row">Joseph</th>
      <td>Anne</td>
      <td>22</td>
    </tr>
    <tr>
      <th scope="row">Peter</th>
      <td>John</td>
      <td>20</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td colspan="3">Le tableau présente les noms, prénoms et âges des
        étudiants</td>
    </tr>
  </tfoot>
</table>
```

Explication de code :

La balise « **table** » est utilisée pour créer le tableau. La balise « **caption** » est utilisée pour donner une légende ou un titre au tableau. Ici, le titre est "Liste des étudiants". La balise « **thead** » contient la ligne d'en-tête du tableau avec les balises « **th** » qui définissent les entêtes de colonnes. Chaque en-tête a un attribut **scope="col"** pour indiquer qu'il s'agit d'un en-tête de colonne. La balise « **tbody** » englobe les lignes du corps du tableau qui contiennent les données des étudiants. Chaque ligne est représentée par la balise « **tr** ». Les données sont incluses dans des balises « **td** ». La première colonne utilise « **th scope="row"** » pour indiquer qu'il s'agit d'une cellule d'en-tête de ligne. La balise « **tfoot** » est utilisée pour le pied du tableau. Dans cet exemple, il y a une seule ligne dans le pied du tableau qui contient une cellule qui s'étend sur les trois colonnes à l'aide de **colspan="3"**.

Ce code crée donc un tableau structuré avec une légende, un en-tête, un corps et un pied de tableau pour afficher la liste des étudiants avec leurs informations.

Tableau 07 : exemple d'un tableau accessible : Liste des étudiants

Nom	Prénom	Âge
Joseph	Anne	22
Peter	John	20

Ce tableau présente les noms, prénoms et âges des étudiants

Remarque : ces balises, permettent de créer une structure de tableau claire pour les technologies d'assistance. On ajoute des informations supplémentaires pour améliorer la compréhension du tableau, telles que la description du tableau dans la balise « **tfoot** ».

3.14 Formulaire accessible

Les formulaires permettent aux utilisateurs d'interagir avec le site et de fournir des informations. Quelques bonnes pratiques à suivre pour créer des formulaires accessibles :

✓ Associer l'étiquette à la zone d'édition

Chaque zone d'édition doit être associée à une étiquette descriptive à l'aide de l'attribut « for » pour que les utilisateurs sachent quelle information doit être saisie. L'étiquette doit être claire et concise pour être facilement compréhensible.

Code 27 : exemple de code HTML, pour associer l'étiquette à la zone d'édition

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

Explication du code :

Les étiquettes « label » sont associées aux zones d'édition en utilisant l'attribut for. Cet attribut est lié à l'ID de la zone de texte, qui est « nom » dans ce cas. Le texte « Nom : » sera donc affiché à côté de la zone de saisie, permettant à l'utilisateur de savoir ce qu'il est censé saisir.

La deuxième ligne est une zone de texte de saisie input qui a un type de « text » et un id de « nom ». Le nom de cette zone de texte est également « nom », ce qui permet aux données saisies d'être transmises au serveur web lors de la soumission du formulaire.

✓ Fournir des exemples de données attendues

Pour aider les utilisateurs à remplir correctement un formulaire, vous pouvez fournir des exemples de données attendues pour chaque champ de formulaire. Cela peut aider les utilisateurs à comprendre ce que vous attendez d'eux.

Code 28 : exemple de code HTML, pour fournir des exemples de données attendues

```
<label for="telephone">Téléphone :</label>
<input type="tel" id="telephone" name="telephone" placeholder="Exemple: 0604162398">
```

Explication du code :

La balise **label** permet de créer l'étiquette qui sera associée au champ de saisie correspondant grâce à l'attribut **for**. Ici, l'attribut for de la balise label est associé à l'attribut id de l'élément input correspondant au champ de saisie de numéro de téléphone. L'élément input de type text permet de créer le champ de saisie. L'attribut id permet de lier cet élément à l'étiquette correspondante grâce à l'attribut for de la balise label. L'attribut name permet de définir le nom de la donnée saisie, qui sera envoyée lors de la soumission du formulaire.

L'attribut placeholder permet d'afficher un exemple de donnée attendue dans le champ de saisie, ici un exemple de format pour le numéro de téléphone. Cela peut aider l'utilisateur à comprendre le type de donnée attendue dans le champ de saisie.

✓ Indiquer les erreurs sous forme de texte

Lorsqu'un utilisateur soumet un formulaire avec des erreurs, vous devez indiquer clairement les erreurs à l'utilisateur. L'utilisation de messages d'erreur textuels clairs et concis est essentielle pour aider les utilisateurs à comprendre ce qui ne va pas.

Code 29 : exemple de code HTML, pour indiquer les erreurs sous forme de texte

```
<label for="email">E-mail :</label>
<input type="email" id="email" name="email" required>
<span class="error">Veuillez entrer une adresse e-mail valide.</span>
```

Explication du code :

La balise « label for= "email " » E-mail : « /label » crée une étiquette "E-mail" qui sera associée à la zone d'édition grâce à l'attribut « for » dont la valeur est l'identifiant de la zone d'édition correspondante. « input type= "email" id= "email" name= "email" required » est la zone d'édition pour saisir l'adresse e-mail. L'attribut "type" est défini sur "email" pour indiquer au navigateur qu'il s'agit d'une adresse e-mail. L'attribut "id" et "name" sont utilisés pour identifier la zone d'édition. L'attribut "required" est utilisé pour indiquer que la saisie est obligatoire. « span class="error" » Veuillez entrer une adresse e-mail valide. « /span » est un élément "span" qui affichera un message d'erreur si l'utilisateur saisit une adresse e-mail invalide.

Remarque : en utilisant cette approche, le formulaire est accessible aux utilisateurs qui utilisent un lecteur d'écran car chaque zone d'édition est associée à une étiquette et les messages d'erreur sont clairement indiqués.

✓ Utiliser des contrôles de formulaire standard

Utilisez des contrôles de formulaire standard tels que les champs de texte, les cases à cocher et les boutons radio pour aider les utilisateurs à comprendre comment remplir votre formulaire.

Code 30 : exemple de code HTML, pour indiquer les erreurs sous forme de texte

```
<label for="choix1"><input type="radio" id="choix1" name="choix" value="1">Choix 1</label>
<label for="choix2"><input type="radio" id="choix2" name="choix" value="2">Choix 2</label>
```

Explication du code :

Les balises « label » sont utilisées pour associer le texte des choix à leur bouton radio correspondant, en utilisant l'attribut for qui a la même valeur que l'attribut id du bouton radio correspondant. Ainsi, lorsque l'utilisateur clique sur le texte du choix, cela sélectionne automatiquement le bouton radio correspondant.

Dans cet exemple, il y a deux choix possibles : "Choix 1" et "Choix 2". Chaque choix a un bouton radio correspondant, identifié par les attributs id et name. Lorsque l'utilisateur sélectionne

l'une des options, le bouton radio correspondant envoie sa valeur (1 pour "Choix 1" et 2 pour "Choix 2") lors de la soumission du formulaire. À noter que les balises « input » doivent être incluses dans les balises « label » pour s'assurer que le texte du choix soit correctement associé au bouton radio correspondant pour une accessibilité optimale.

Remarque : en suivant ces bonnes pratiques, vous pouvez créer des formulaires accessibles pour tous les utilisateurs, y compris ceux qui utilisent des technologies d'assistance telles que les lecteurs d'écran et les claviers uniquement.

3.15 Restituer les composants interactifs

Lors de la conception d'une interface web, il est important de s'assurer que les composants interactifs sont correctement restitués par les lecteurs d'écran.

Quelques conseils pour rendre ces éléments accessibles :

✓ **Utiliser des balises HTML sémantiques**

Les balises HTML sémantiques telles que « nav », « header », « main » et « footer » sont importantes pour la navigation sur les pages web. Les lecteurs d'écran utilisent ces balises pour naviguer dans le contenu de la page.

✓ **Fournir une description textuelle**

Les images, les graphiques et les icônes doivent avoir une description textuelle qui peut être lue par les lecteurs d'écran. Cette description doit décrire l'image de manière concise et précise.

✓ **Utiliser des étiquettes pour les éléments interactifs**

Les boutons, les liens, les cases à cocher, les boutons radio et les zones de texte doivent être étiquetés correctement pour permettre aux utilisateurs de comprendre leur fonction.

✓ **Éviter les éléments interactifs non standards**

Les éléments interactifs non standards tels que les menus déroulants personnalisés, les boutons de sélection de date et les curseurs personnalisés peuvent ne pas être accessibles aux lecteurs d'écran. Il est préférable d'utiliser des éléments standards pour lesquels l'accessibilité est garantie.

✓ **Tester avec des lecteurs d'écran**

Il est important de tester l'interface avec des lecteurs d'écran pour s'assurer que les éléments interactifs sont correctement restitués.

✓ **Utiliser des instructions claires**

Les instructions pour les éléments interactifs tels que les menus, les barres de progression et les calendriers doivent être claires et concises.

Remarque : en suivant ces conseils, vous pouvez rendre les composants interactifs de votre site web accessibles aux utilisateurs de lecteurs d'écran.

3.16 Séparation du fond et la forme

La séparation du fond et de la forme est un principe important pour la conception de pages Web efficaces et accessibles. La séparation du fond et de la forme est illustrée par l'utilisation de la balise « link » dans l'en-tête du HTML pour relier la page à un fichier CSS externe. Toutes les instructions de présentation sont définies dans ce fichier CSS, ce qui permet de les modifier facilement sans toucher au contenu de la page.

Le contenu de la page est écrit en langage HTML, qui décrit la structure et le sens du contenu de la page. Le HTML permet de définir les balises qui indiquent les titres, les paragraphes, les listes, les images, les liens, etc. Cela permet aux navigateurs et aux technologies d'assistance de comprendre la structure de la page et d'aider les utilisateurs à naviguer dans le contenu.

La présentation de la page est définie en langage CSS, qui décrit l'apparence visuelle de la page. Le CSS permet de définir les styles qui contrôlent la police, la taille, la couleur, l'espacement, le positionnement, etc.

Ces règles sont souvent régies par des normes d'accessibilité numérique telles que les normes WCAG (Web Content Accessibility Guidelines), qui sont largement reconnues et utilisées dans le monde entier.

4 Accessibilité des documents bureautiques⁶

L'accessibilité des documents bureautiques est un concept important pour permettre aux personnes en situation de handicap d'accéder aux informations contenues dans ces documents. Pour les fichiers au format Word, PDF, les principes sont presque identiques que le web.

4.1 Utiliser une structure de document cohérente

Organisez votre document en utilisant une structure logique de titres et de sous-titres, qui reflète la hiérarchie de l'information. Utilisez les styles de titre de Word pour donner une hiérarchie aux titres, plutôt que de mettre en gras ou en italique.

Les étapes pour utiliser les styles de titre dans Word

1. Ouvrez un document Word existant ou créez-en un nouveau.
2. Sélectionnez un titre auquel vous souhaitez appliquer un style. Par exemple, "Introduction".
3. Dans l'onglet "Accueil", cliquez sur le bouton "Styles" pour afficher la galerie de styles.
4. Choisissez un style de titre approprié pour le titre sélectionné. Par exemple, le style "Titre 1".
5. Répétez l'opération pour chaque titre de niveau supérieur dans votre document.
6. Pour les sous-titres, utilisez les styles de titre de niveau inférieur, comme "Titre 2", "Titre 3", etc.

Remarque : Les styles de Word permettent aux technologies d'assistance de comprendre la structure logique du document, ce qui facilite la navigation et la lecture pour les utilisateurs ayant des troubles visuels ou cognitifs.

Exemple de structuration cohérente pour un rapport de projet

1. Introduction

- 1.1. Objectifs du projet
- 1.2. Contexte du projet
- 1.3. Portée du rapport

2. Description du projet

- 2.1. Description détaillée du projet
- 2.2. Méthodologie de travail utilisée

4.2 Utiliser des couleurs contrastées

Assurez-vous que le contraste des couleurs utilisées dans le document est suffisant pour être visible par tous, y compris les personnes atteintes de daltonisme.

⁶ https://disic.github.io/guides-documents_bureautiques_accessible/html/

Pour le texte en noir, utiliser un fond blanc ou très clair pour créer un contraste élevé.

Exemple de couleurs contrastées

Titres en noir sur fond blanc

Sous-titres en bleu foncé sur fond blanc

Texte normal en noir sur fond gris clair

Évitez les combinaisons de couleurs qui ont un faible contraste, telles que le rouge foncé sur un fond vert foncé.

Le rouge foncé sur un fond vert

4.3 Utiliser une police facile à lire

Utilisez une police sans-serif comme Arial, Verdana, Luciole ou Calibri, avec une taille de police suffisamment grande (au moins 12 points) pour être lisible par tous. Il est recommandé de maintenir une taille de police d'au moins 12 points pour le texte normal et de 14 à 16 points pour les titres et sous-titres.

Exemple de choix de police pour un document bureautique

Titres en Calibri Bold 16

Sous-titres en Arial Bold 14

Texte normal en Arial 12

4.4 Ajouter des descriptions alternatives aux images

Pour les images, ajoutez une description alternative qui décrit l'image de manière concise. Ceci est particulièrement important pour les personnes aveugles ou malvoyantes.

Les étapes pour ajouter des descriptions alternatives aux images dans Word

1. Cliquez sur l'image que vous souhaitez décrire pour la sélectionner.
2. Cliquez avec le bouton droit de la souris sur l'image et sélectionnez "Format de l'image" dans le menu déroulant.
3. Dans la boîte de dialogue qui apparaît, sélectionnez l'onglet "Texte alternatif".
4. Dans la zone de texte "Description", saisissez une description concise mais précise de l'image. Assurez-vous d'inclure toutes les informations importantes de l'image.
5. Cliquez sur OK pour enregistrer la description alternative de l'image.

Exemple de description alternative pour une image dans un document bureautique

Nom de l'image : "Graphique des ventes mensuelles."

Description alternative : "Graphique montrant les ventes mensuelles de l'entreprise pour l'année en cours. Les ventes ont augmenté régulièrement au cours des six derniers mois."

4.5 Utiliser les tableaux de manière appropriée

Utilisez les tableaux uniquement pour présenter des données tabulaires. Assurez-vous que les tableaux sont clairement structurés et facilement compréhensibles par tous.

Les étapes pour rendre un tableau accessible dans un document bureautique

1. Utilisez la fonction "Tableau" de votre logiciel bureautique pour créer le tableau.
2. Définir les lignes de titre pour identifier les en-têtes de tableau. Les lignes de titre doivent être situées en haut du tableau et contenir les titres de chaque colonne.
3. Utilisez les en-têtes de tableau pour indiquer le contenu de chaque colonne. Les en-têtes doivent être clairs et concis, et ils doivent décrire le contenu de la colonne de manière précise.
4. Assurez que les couleurs utilisées dans le tableau ont suffisamment de contraste pour être visibles par tous.

Exemple de tableau accessible dans un document bureautique

Liste des employés d'Imago le réseau			
Nom	Prénom	Âge	Sexe
Jean	Philippe	26	Homme
Joseph	Marie	30	Femme

Ce tableau présente les noms, prénoms, âges et les sexe des employés

4.6 Utiliser les listes à puces

Utiliser des listes à puces dans un document bureautique permet de structurer l'information en listant des éléments de manière claire et concise.

Les étapes à suivre Pour créer une liste à puces dans un document bureautique

1. Sélectionnez le texte que vous souhaitez inclure dans la liste à puces.
2. Dans l'onglet "Accueil", cliquez sur l'icône "Liste à puces" dans la barre d'outils.
3. Le texte sélectionné est maintenant dans une liste à puces. Répétez l'opération pour chaque élément de la liste.

Exemple de liste à puces

- Manger du fromage.
- Manger du pain.
- Boire du lait.

Remarque : pour rendre une liste à puces accessible, il est recommandé d'utiliser une puce simple plutôt qu'une image ou un caractère spécial. Il est important d'utiliser une structure cohérente et de commencer chaque élément par une majuscule et de terminer par un point ou une virgule. Enfin, évitez de créer des listes à puces trop longues ou complexes, car cela peut rendre la lecture difficile pour certaines personnes.

4.7 Ajouter des liens

Ajouter des liens dans un document bureautique peut aider à rendre le document plus accessible et à faciliter la navigation pour les utilisateurs.

Les étapes pour ajouter des liens dans un document Word

1. Sélectionnez le texte ou l'image à laquelle vous voulez ajouter un lien.
2. Cliquez sur l'onglet "Insertion" dans le ruban en haut de la fenêtre.
3. Cliquez sur l'icône "Lien" dans le groupe "Liens".
4. Dans la boîte de dialogue qui s'ouvre, saisissez ou collez l'URL du lien dans le champ "Adresse". Par exemple, "<https://www.imagolereseau.fr/>".
5. Ajoutez une description concise et précise du lien dans le champ "Texte à afficher". Par exemple, "[Visitez la page d'accueil d'Imago Le Réseau pour en savoir plus](#)".
6. Cliquez sur "OK" pour enregistrer le lien.
7. Testez le lien en cliquant dessus pour vous assurer qu'il fonctionne correctement.

Remarque : Il est important de noter que les liens doivent être clairement identifiables pour les utilisateurs. Évitez d'utiliser des phrases ambiguës comme "Cliquez ici" ou "Lire la suite" comme texte de lien. Il est préférable d'utiliser une description du lien qui donne une idée précise du contenu de la page liée.

4.8 Vérifier l'accessibilité du document Word

Utilisez l'outil "Vérification d'accessibilité" de Word pour vérifier si votre document respecte les critères d'accessibilité.

Les étapes pour vérifier l'accessibilité d'un document Word

1. Ouvrez le document Word que vous souhaitez vérifier.
2. Dans l'onglet "Fichier", cliquez sur " info" dans le menu déroulant.
3. Dans la fenêtre qui ouvre, cliquez sur " vérifier l'absence de problèmes"
4. Dans la boîte de dialogue qui s'ouvre, cochez la case "Vérifier l'accessibilité".
5. Cliquez sur "Démarrer" pour commencer la vérification.
6. Après la vérification, des erreurs et des avertissements éventuelles s'afficheront dans la boîte de dialogue "Vérification de l'accessibilité".
7. Cliquez sur chaque erreur ou avertissement pour obtenir des informations détaillées sur les problèmes et les suggestions pour les corriger.
8. Corrigez chaque erreur ou avertissement en suivant les suggestions données.
9. Vérifiez à nouveau le document après avoir apporté des modifications.

Remarque : il est important de s'assurer que le document suit les principes de l'accessibilité numérique tels que définis par les WCAG 2.1, ainsi que d'effectuer des tests d'utilisabilité pour garantir que le document est facilement utilisable pour tous les utilisateurs.

4.9 Accessibilité d'un document PDF

L'accessibilité d'un document PDF est essentielle pour permettre à tous les utilisateurs de le comprendre et de l'utiliser, y compris les personnes atteintes de handicaps visuels ou moteurs.

Quelques étapes pour rendre un document PDF accessible

1. Utilisez un logiciel de création de PDF qui prend en charge les normes d'accessibilité telles que Adobe Acrobat Pro DC.

Adobe Acrobat Pro DC⁷ offre des fonctionnalités d'accessibilité pour aider à rendre les documents PDF accessibles aux personnes atteintes de handicaps visuels, auditifs ou physiques. Il permet notamment de :

- ✓ Ajouter des descriptions alternatives pour les images, les graphiques et les tableaux.
- ✓ Créer des étiquettes et champs de formulaire pour faciliter la navigation au clavier.
- ✓ Ajouter des balises pour améliorer la structure logique du document et la lisibilité pour les lecteurs d'écran.
- ✓ Vérifier l'accessibilité du document PDF et corriger les erreurs ou les avertissements.

Ou

1. Créez un document Word en utilisant les principes d'accessibilité, tels que l'utilisation de titres, de tableaux de données, de listes à puces, etc.

4.10 Vérifier l'accessibilité du PDF

PDF Accessibility Checker est un outil en ligne gratuit qui permet de vérifier l'accessibilité d'un document PDF, cet outil est disponible en plusieurs langues.

PDF Accessibility Checker peut être utilisé pour vérifier si un document PDF est conforme aux normes d'accessibilité telles que les WCAG⁸. L'outil fournit des recommandations pour corriger les erreurs détectées.

Pour utiliser PDF Accessibility Checker⁹, il suffit de télécharger le document PDF à vérifier sur le site web de l'outil, puis de lancer l'analyse. Les résultats s'affichent sous forme de rapports détaillés, avec des informations sur les erreurs, les avertissements et les recommandations pour améliorer l'accessibilité du document.

⁷ Ce logiciel est payant

⁸ Web Content Accessibility Guidelines

⁹ <https://www.abracadabrapdf.net/ressources-et-tutos/edition-modification/pdf-accessibility-checker-2021/>

4.11 Table des matières accessible

La table des matières joue un rôle important en matière d'accessibilité dans un document bureautique. Elle permet aux utilisateurs de naviguer rapidement et efficacement dans le document en identifiant les différentes sections et en accédant directement aux parties qui les intéressent. Cela est particulièrement bénéfique pour les personnes ayant une déficience visuelle ou celles qui utilisent des technologies d'assistance pour accéder au contenu.

Quelques bonnes pratiques à suivre pour rendre une table des matières accessible :

1. Appliquez les styles de titre appropriés dans votre document bureautique pour créer une hiérarchie claire des sections. Cela permettra à la table des matières de refléter la structure du document. [Comment le faire ?](#)
2. Utilisez les fonctionnalités de génération automatique de la table des matières disponibles dans les logiciels de traitement de texte. Cela garantit que la table des matières est toujours à jour et synchronisée avec le contenu réel du document.

Suivez les étapes suivantes :

- Utilisez des styles tels que "Titre 1" pour les titres principaux, "Titre 2" pour les sous-titres, et ainsi de suite.
 - Placez le curseur à l'endroit où vous souhaitez insérer la table des matières.
 - Dans l'onglet "Références" ou "Insertion". Recherchez l'option "Table des matières" ou "Génération automatique de la table des matières".
 - Une fois que vous avez accédé à la fonction table des matières, vous pouvez choisir parmi différents styles de table des matières prédéfinis.
 - Cliquez sur le bouton "Générer" ou "Insérer" pour créer automatiquement la table des matières dans votre document.
3. Veillez à ce que la mise en forme des entrées de la table des matières soit cohérente, notamment en termes de police, de taille de texte et de couleurs. Utilisez des styles prédéfinis pour maintenir une apparence uniforme. [Comment le faire ?](#)
 4. Assurez-vous que les entrées de la table des matières sont des liens cliquables, permettant aux utilisateurs de naviguer directement vers les sections correspondantes du document.

Suivez les étapes suivantes :

- Une fois la table des matières générée, vérifiez que chaque entrée est un lien cliquable. Vous devriez pouvoir cliquer sur une entrée de la table des matières pour accéder directement à la section correspondante du document.
 - Testez les liens : Effectuez des tests en cliquant sur chaque entrée de la table des matières pour vous assurer que les liens fonctionnent correctement et vous amènent à la bonne section du document.
5. Fournir des descriptions alternatives aux entrées de la table des matières. Cela permet aux lecteurs d'écran de comprendre le contenu de chaque section. [Comment le faire ?](#)

5 QCM sur l'accessibilité numérique

1. Qu'est-ce que l'accessibilité numérique ?

- a. L'ensemble des techniques pour améliorer le référencement d'un site web
- b. L'ensemble des règles et des bonnes pratiques pour rendre un site web utilisable par tous, y compris les personnes en situation de handicap
- c. L'ensemble des procédés pour rendre un site web interactif et attractif pour l'utilisateur

Rep :

2. Quels sont les types de handicap pris en compte dans l'accessibilité numérique ?

- a. Seulement les personnes malvoyantes
- b. Seulement les personnes sourdes
- c. Tous les types de handicap (moteur, visuel, auditif, cognitif, mental etc.)

Rep :

3. A quoi correspond le contraste des couleurs dans l'accessibilité numérique ?

- a. La différence de couleur entre le texte et le fond
- b. L'utilisation de couleurs vives pour attirer l'attention de l'utilisateur
- c. La variété de couleurs utilisées pour le design du site

Rep :

4. Comment rendre un site web utilisable au clavier ?

- a. En ne proposant que des liens cliquables
- b. En utilisant des raccourcis clavier pour chaque élément
- c. En permettant la navigation dans le site en utilisant la touche Tabulation

Rep :

5. Qu'est-ce que le RGAA 4.1 ?

- a. Un standard international pour l'accessibilité numérique
- b. Un référentiel français pour l'accessibilité numérique
- c. Un outil de test d'accessibilité pour les développeurs web

Rep :

6. Quel est l'objectif du RGAA 4.1 ?

- a. Garantir que les sites web sont conformes aux normes d'accessibilité internationales
- b. Améliorer l'accessibilité numérique des sites web en France
- c. Tester l'accessibilité des sites web en utilisant des outils automatisés

Rep :

7. Qu'est-ce que le WCAG 2.1 ?

- a. Un standard international pour l'accessibilité numérique
- b. Un référentiel français pour l'accessibilité numérique
- c. Un outil de test d'accessibilité pour les développeurs web

Rep :

8. Quel est l'objectif du WCAG 2.1 ?

- a. Garantir que les sites web sont conformes aux normes d'accessibilité internationales
- b. Améliorer l'accessibilité numérique des sites web en France
- c. Tester l'accessibilité des sites web en utilisant des outils automatisés

Rep :

9. Quelles sont les différences entre le RGAA 4.1 et le WCAG 2.1 ?

- a. Le RGAA 4.1 est un référentiel français tandis que le WCAG 2.1 est un standard international
- b. Le RGAA 4.1 est basé sur le WCAG 2.1 mais inclut des critères spécifiques à la France
- c. Le RGAA 4.1 est plus restrictif que le WCAG 2.1

Rep :

10. Quelles sont les niveaux de conformité du WCAG 2.1 ?

- a. A, AA et AAA
- b. Bronze, Argent et Or
- c. Faible, Moyen et Élevé

Rep :

11. Quel niveau de conformité est recommandé par le RGAA 4.1 ?

- a. Niveau A
- b. Niveau AA
- c. Niveau AAA

Rep :

12. Que signifie le test de contraste de couleur ?

- a. Tester si les couleurs de l'interface utilisateur sont agréables à l'œil
- b. Tester si les couleurs de l'interface utilisateur sont conformes aux normes d'accessibilité
- c. Tester si les couleurs de l'interface utilisateur sont compatibles avec les navigateurs web

Rep :

13. Pourquoi est-il important d'associer une étiquette à un champ de formulaire ?

- a. Pour des raisons de sécurité
- b. Pour que le champ soit affiché correctement sur tous les navigateurs
- c. Pour faciliter la navigation et l'utilisation du formulaire par les personnes utilisant un lecteur d'écran ou une assistance technique
- d. Pour ajouter une description supplémentaire au champ

Rep :

Sites Internet de référence

- Acrobat du format PDF : [PDF Accessibility Checker 2021](#)
- Support Microsoft : [Améliorer l'accessibilité à l'aide du Vérificateur d'accessibilité](#)
- Web Accessibility Initiative : [Introduction à l'accessibilité du web](#)
- Web Accessibility Initiative : [Principes fondamentaux de l'accessibilité - Vue d'ensemble](#)
- Le Référentiel [général d'amélioration de l'accessibilité](#) (RGAA)
- Web Content [Accessibility Guidelines](#) (WCAG) 2.1
- Créer des [documents bureautiques accessibles](#)
- Vérifier le contraste du web : [Colorable jxnblk](#)
- Web accessibility in mind : [webaim contrast checker](#)
- Colour Contrast Checker : [colour contrast](#)

Des Ressources à consulter

- "10 principes pour une accessibilité numérique universelle" de la Fédération des Aveugles et Amblyopes de France : https://www.avh.asso.fr/sites/default/files/inline-files/10_principes_accessibilite_num.pdf
- "Guide de l'accessibilité numérique" de l'Association Valentin Haüy : https://www.avh.asso.fr/sites/default/files/inline-files/Guide_Accessibilite_Num.pdf
- "Guide pratique de l'accessibilité numérique" du CNAM : <https://www.cnam.fr/ressources-liens/accessibilite-numerique-1197463.kjsp>
- "L'accessibilité numérique en 10 points clés" de la Caisse Nationale de Solidarité pour l'Autonomie : https://solidarites-sante.gouv.fr/IMG/pdf/10_points_cles_accessibilite_numerique_cnaiep.pdf
- "Manuel d'accessibilité web" du W3C : <https://www.w3.org/WAI/standards-guidelines/wcag/glance/>
- "Ressources sur l'accessibilité numérique" de l'Agence du numérique : <https://www.numerique.gouv.fr/accessibilite/ressources/>

Remarque : Toutes ces ressources sont gratuites et disponibles en format PDF sur les sites correspondants.

Contact

Elodie CHASSAING

Responsable Pôle Art & Handicap 78/92

Membre actif d'IMAGO-Le réseau

echassaing@seay.fr

&

Michelet LAMOUR

Spécialiste en Technologie et Handicap

« Accessibilité Numérique »

lamourmichelet37@gmail.com