
WEB Dynamique

Version 0.0.1

Emery Assogba

oct. 05, 2021

Contents:

1	HTML, XHTML et CSS	1
1.1	Introduction	1
1.2	Qu'est-ce que le HTML ?	1
1.3	Qu'est-ce que XHTML ?	2
1.4	Qu'est-ce que le CSS ?	2
1.5	Qu'est-ce que WebFonts ?	2
1.6	Exemples	2
2	JAVASCRIPT WEB APIS	3
2.1	Qu'est-ce que le scripting ?	3
2.2	Quelles interfaces de script sont disponibles ?	3
2.3	Au-delà des scripts	4
3	GRAPHIQUES	5
3.1	Qu'est-ce que les graphiques ?	5
3.2	A quoi servent les graphiques ?	6
3.3	Qu'est-ce que le PNG ?	6
3.4	Qu'est-ce que le SVG ?	6
3.5	Qu'est-ce que le CSS ?	6
3.6	Qu'est-ce que l'API Canvas ?	7
3.7	Qu'est-ce que WebCGM ?	7
3.8	Exemples	7
4	Audio et Vidéo	9
4.1	Qu'est-ce que l'audio et la vidéo ?	9
4.2	A quoi servent l'audio et la vidéo ?	9
4.3	Qu'est-ce que SMIL ?	10
4.4	Qu'est-ce que le texte chronométré ?	10
4.5	Que sont les fragments de médias ?	10
4.6	Que sont les annotations multimédias ?	10
4.7	Exemples	10
5	Accessibilité	13
5.1	Pourquoi : les arguments en faveur de l'accessibilité du Web	13
5.2	Quoi : exemples d'accessibilité Web	14
5.3	Comment : rendre votre site Web et vos outils Web accessibles	14

6	Internationalisation	17
6.1	Qu'est-ce que l'internationalisation ?	17
6.2	Exemples	17
7	PHP	21
7.1	Qu'est-ce que PHP ?	21
7.2	Qu'est-ce qu'un fichier PHP ?	21
7.3	Que peut faire PHP ?	21
7.4	Pourquoi PHP ?	22
7.5	Quoi de neuf en PHP 7	22
7.6	Installation de PHP	22
8	Syntaxe de PHP	23
8.1	Syntaxe PHP de base	23
8.2	Commentaires en PHP	24
8.3	Les formulaires en PHP	26
9	Indices and tables	29

1.1 Introduction

HTML (Hypertext Markup Language) et CSS (Cascading Style Sheets) sont deux des technologies de base pour la création de pages Web. HTML fournit la structure de la page, CSS la mise en page (visuelle et auditive), pour une variété d'appareils. Avec les graphiques et les scripts, HTML et CSS sont à la base de la création de pages Web et d'applications Web.

1.2 Qu'est-ce que le HTML ?

HTML est le langage utilisé pour décrire la structure des pages Web. HTML donne aux auteurs les moyens de :

- Publiez des documents en ligne avec des titres, du texte, des tableaux, des listes, des photos, etc.
- Récupérez des informations en ligne via des liens hypertextes, d'un simple clic.
- Concevoir des formulaires pour effectuer des transactions avec des services à distance, à utiliser pour rechercher des informations, effectuer des réservations, commander des produits, etc.
- Inclure des feuilles de calcul, des clips vidéo, des clips audio et d'autres applications directement dans leurs documents.

Avec HTML, les auteurs décrivent la structure des pages en utilisant le balisage (markup). Les éléments du langage étiquettent les éléments de contenu tels que «paragraphe», «liste», «tableau», etc.

1.3 Qu'est-ce que XHTML ?

XHTML est une variante de HTML qui utilise la syntaxe de XML, le langage de balisage extensible. XHTML a tous les mêmes éléments (pour les paragraphes, etc.) que la variante HTML, mais la syntaxe est légèrement différente. Étant donné que XHTML est une application XML, vous pouvez utiliser d'autres outils XML (comme XSLT, un langage pour transformer le contenu XML).

1.4 Qu'est-ce que le CSS ?

CSS est le langage utilisé pour décrire la présentation des pages Web, y compris les couleurs, la mise en page et les polices. Il permet d'adapter la présentation à différents types d'appareils, tels que les grands écrans, les petits écrans ou les imprimantes. CSS est indépendant du HTML et peut être utilisé avec n'importe quel langage de balisage basé sur XML. La séparation du HTML du CSS facilite la maintenance des sites, le partage des feuilles de style entre les pages et l'adaptation des pages à différents environnements. C'est ce qu'on appelle la séparation de la structure (ou : du contenu) de la présentation.

1.5 Qu'est-ce que WebFonts ?

WebFonts est une technologie qui permet aux utilisateurs d'utiliser des polices à la demande sur le Web sans nécessiter d'installation dans le système d'exploitation. Le W3C a de l'expérience dans les polices téléchargeables via HTML, CSS2 et SVG. Jusqu'à récemment, les polices téléchargeables n'étaient pas courantes sur le Web en raison de l'absence d'un format de police interopérable. L'effort de WebFonts prévoit de résoudre ce problème en créant un format de police ouvert et pris en charge par l'industrie pour le Web (appelé « WOFF »).

1.6 Exemples

L'exemple très simple suivant d'une partie d'un document HTML illustre comment créer un lien dans un paragraphe. Lorsqu'il est rendu à l'écran (ou par un synthétiseur vocal), le texte du lien sera le « rapport final » ; lorsque quelqu'un active le lien, le navigateur récupère la ressource identifiée par « <http://www.example.com/report> » :

```
<p class="moreinfo">Pour plus d'information consultez le  
<a href="http://www.example.com/report">rapport final</a>.</p>
```

L'attribut class sur la balise de début du paragraphe (« <p> ») peut être utilisé, entre autres, pour ajouter du style. Par exemple, pour mettre en italique le texte de tous les paragraphes avec une classe de « moreinfo », on pourrait écrire, en CSS :

```
p.moreinfo { font-style: italic }
```

En plaçant cette règle dans un fichier séparé, le style peut être partagé par plusieurs documents HTML.

Pour plus d'information visitez http://slaout.linux62.org/html_css/ et <https://www.w3.org/Style/Examples/011/firstcss>

2.1 Qu'est-ce que le scripting ?

Un script est un programme qui n'a pas besoin de pré-traitement (par exemple, compilation) avant d'être exécuté. Dans le contexte d'un navigateur Web, les scripts font généralement référence au code de programme écrit en JavaScript qui est exécuté par le navigateur lorsqu'une page est téléchargée ou en réponse à un événement déclenché par l'utilisateur.

Les scripts peuvent rendre les pages Web plus dynamiques. Par exemple, sans recharger une nouvelle version d'une page, il peut autoriser des modifications du contenu de cette page (DHTML (Dynamic HTML)), ou autoriser l'ajout ou l'envoi de contenu à partir de cette page (AJAX (Asynchronous JavaScript and XML)).

Au-delà, les scripts permettent de plus en plus aux développeurs de créer un pont entre le navigateur et la plateforme sur laquelle il s'exécute, permettant, par exemple, de créer des pages Web qui intègrent des informations de l'environnement de l'utilisateur, telles que l'emplacement actuel, les détails du carnet d'adresses, etc.

Cette interactivité supplémentaire fait que les pages Web se comportent comme une application logicielle traditionnelle. Ces pages Web sont souvent appelées applications Web et peuvent être mises à disposition soit directement dans le navigateur sous forme de page Web, soit empaquetées et distribuées sous forme de widgets.

2.2 Quelles interfaces de script sont disponibles ?

L'interface de script la plus basique développée au W3C est le DOM, le Document Object Model qui permet aux programmes et aux scripts d'accéder et de mettre à jour dynamiquement le contenu, la structure et le style des documents. Les spécifications DOM forment le cœur du DHTML.

Les modifications du contenu à l'aide du DOM par l'utilisateur et par des scripts déclenchent des événements que les développeurs peuvent utiliser pour créer des interfaces utilisateur riches.

Un certain nombre d'interfaces plus avancées sont en cours de standardisation, par exemple :

- XMLHttpRequest permet de charger du contenu supplémentaire à partir du Web sans charger un nouveau document, un composant central d'AJAX,
- l'API de géolocalisation rend l'emplacement actuel de l'utilisateur disponible pour les applications basées sur un navigateur,

- plusieurs API rendent transparente l'intégration des applications Web avec le système de fichiers local et le stockage.

WAI ARIA propose des mécanismes pour garantir que cette interactivité supplémentaire reste utilisable indépendamment des appareils et des handicaps. Des considérations supplémentaires s'appliquent au développement d'applications Web pour appareils mobiles.

2.3 Au-delà des scripts

Alors que les scripts offrent une excellente opportunité de développer de nouvelles interfaces et d'expérimenter de nouvelles interactions avec les utilisateurs, au fil du temps, un certain nombre de ces ajouts bénéficient d'une approche plus déclarative ; par exemple, au lieu que chaque développeur réimplémente une interface de calendrier qui permet à un utilisateur de choisir une date, définir un type d'entrée (`<input type="date" />`) qui le fait automatiquement permet de gagner beaucoup de temps et les bugs, et crée un terrain pour de nouvelles innovations.

Au-delà de l'ensemble des interfaces déclaratives mises à disposition via HTML, plusieurs technologies ont été développées pour rendre possibles ces applications Web déclaratives.

Pour plus d'information visitez les liens suivants : https://developer.mozilla.org/fr/docs/Learn/Getting_started_with_the_web/JavaScript_basics , <https://developer.mozilla.org/fr/docs/Web/JavaScript/Guide/Introduction> , https://developer.mozilla.org/fr/docs/Learn/Tools_and_testing/Client-side_JavaScript_frameworks et https://developer.mozilla.org/fr/docs/Web/JavaScript/Inheritance_and_the_prototype_chain

Le Web, c'est plus que du texte et de l'information, c'est aussi un moyen d'expression de la créativité artistique, de la visualisation de données et d'optimisation de la présentation de l'information pour différents publics avec des besoins et des attentes différents. L'utilisation de graphiques sur les sites Web améliore l'expérience des utilisateurs, et le W3C dispose de plusieurs technologies différentes et complémentaires qui fonctionnent avec HTML et les scripts pour fournir aux créateurs de pages Web et d'applications Web les outils dont ils ont besoin pour fournir la meilleure représentation possible de leur contenu.

3.1 Qu'est-ce que les graphiques ?

Les graphiques Web sont des représentations visuelles utilisées sur un site Web pour améliorer ou permettre la représentation d'une idée ou d'un sentiment, afin d'atteindre l'utilisateur du site Web. Les graphiques peuvent divertir, éduquer ou avoir un impact émotionnel sur l'utilisateur et sont essentiels à la force de l'image de marque, à la clarté de l'illustration et à la facilité d'utilisation des interfaces.

Des exemples de graphiques incluent des cartes, des photographies, des conceptions et des modèles, des arbres généalogiques, des diagrammes, des plans architecturaux ou techniques, des graphiques à barres et des camemberts, une typographie, des schémas, des dessins au trait, des organigrammes et de nombreuses autres formes d'images.

Les graphistes disposent de nombreux outils et technologies pour tout, de l'impression au développement Web, et le W3C fournit de nombreux formats sous-jacents pouvant être utilisés pour la création de contenu sur la plate-forme Web ouverte.

3.2 A quoi servent les graphiques ?

Les graphiques sont utilisés pour tout, de l'amélioration de l'apparence des pages Web à la couche de présentation et d'interaction utilisateur pour les applications Web à part entière.

Différents cas d'utilisation des graphiques exigent des solutions différentes, il existe donc plusieurs technologies différentes. Les photographies sont mieux représentées avec PNG, tandis que les dessins au trait interactifs, la visualisation des données et même les interfaces utilisateur ont besoin de la puissance de SVG et de l'API Canvas. CSS existe pour améliorer d'autres formats comme HTML ou SVG. WebCGM répond aux besoins d'illustration et de documentation techniques de nombreuses industries.

3.3 Qu'est-ce que le PNG ?

Portable Network Graphics (PNG) est un format de fichier statique pour le stockage et l'échange sans perte, portable et bien compressé d'images raster (bitmaps). Il offre un contrôle des couleurs riche, avec une prise en charge des couleurs indexées, des niveaux de gris et des couleurs vraies et une transparence du canal alpha. PNG est conçu pour le Web, avec des capacités de streaming et de rendu progressif. Il est pris en charge de manière omniprésente dans les navigateurs Web, les outils de création graphique, les boîtes à outils d'images et d'autres parties de la chaîne d'outils de création. Les fichiers PNG ont l'extension de fichier « .PNG » ou « .png » et doivent être déployés en utilisant le type de média « image/png ». Les images PNG peuvent être utilisées avec HTML, CSS, SVG, l'API Canvas et WebCGM.

3.4 Qu'est-ce que le SVG ?

Les graphiques vectoriels évolutifs (SVG) sont comme le HTML pour les graphiques. Il s'agit d'un langage de balisage permettant de décrire tous les aspects d'une image ou d'une application Web, de la géométrie des formes au style du texte et des formes, en passant par l'animation et les présentations multimédias, y compris la vidéo et l'audio. Il est entièrement interactif et inclut un DOM scriptable ainsi qu'une animation déclarative (via la spécification SMIL). Il prend en charge un large éventail de fonctionnalités visuelles telles que les dégradés, l'opacité, les filtres, l'écartèlement et le masquage.

L'utilisation de SVG permet des graphiques entièrement évolutifs, fluides et réutilisables, des graphiques simples pour améliorer les pages HTML, aux graphiques entièrement interactifs et à la visualisation des données, aux jeux, aux images statiques autonomes de haute qualité. SVG est pris en charge nativement par la plupart des navigateurs modernes (avec des plugins pour permettre son utilisation sur tous les navigateurs) et est largement disponible sur les appareils mobiles et les décodeurs. Tous les principaux outils de dessin de graphiques vectoriels importent et exportent du SVG, et ils peuvent également être générés via des langages de script côté client ou côté serveur.

3.5 Qu'est-ce que le CSS ?

Les feuilles de style en cascade (CSS) sont le langage utilisé pour décrire la présentation des pages Web, y compris les couleurs, la mise en page et les informations sur les polices. Il peut être utilisé pour améliorer les aspects graphiques du HTML et du SVG.

3.6 Qu'est-ce que l'API Canvas ?

L'API Canvas est une technologie de script côté client permettant la création ou la modification d'images raster (bit-maps) . Il utilise des méthodes de programmation vectorielles pour créer des formes, des dégradés et d'autres effets graphiques, et comme il n'a pas de DOM, il peut fonctionner très rapidement. Les scripteurs dédiés peuvent développer des jeux ou même des applications complètes en utilisant l'API Canvas, seule ou intégrée dans HTML ou SVG. Il est pris en charge nativement dans la plupart des navigateurs modernes (avec des bibliothèques de scripts étendant la prise en charge à tous les principaux navigateurs), et même sur certains appareils mobiles.

3.7 Qu'est-ce que WebCGM ?

Web Computer Graphics Metafile (WebCGM) est le profil Web de CGM, la norme ISO pour la définition d'images vectorielles et composites vectorielles/raster. CGM a un nombre important d'adeptes dans l'illustration technique, la documentation électronique interactive, la visualisation de données géophysiques, entre autres domaines d'application, et est largement utilisé dans les domaines de l'ingénierie automobile, de l'aéronautique et de l'industrie de la défense. WebCGM possède bon nombre des mêmes fonctionnalités graphiques que SVG.

3.8 Exemples

Comme exemple simple de fichier image SVG, voici un cercle avec un dégradé pour lui donner un aspect 3D :

```
<svg xmlns="http://www.w3.org/2000/svg" width="100%" height="100%" viewBox="0 0 100
↪100">

  <defs>
    <radialGradient id="rg" cx="100" cy="100" fx="80" fy="80" gradientUnits=
↪"userSpaceOnUse">
      <stop offset="5%" stop-color="lightskyblue" />
      <stop offset="100%" stop-color="darkblue" />
    </radialGradient>
  </defs>
  <circle id="circle_1" cx="100" cy="100" r="95" fill="url(#rg)" />
</svg>
```

Pour plus d'informations visitez <https://www.w3.org/Consortium/Offices/Presentations/SVG/0.svg> ou <https://www.w3.org/TR/SVGPrintPrimer12/>

4.1 Qu'est-ce que l'audio et la vidéo ?

Les termes audio et vidéo font généralement référence au format de stockage multimédia basé sur le temps pour les informations sur le son/la musique et les images animées. L'enregistrement numérique audio et vidéo, également appelé codecs audio et vidéo, peut être non compressé, compressé sans perte ou compressé avec perte en fonction de la qualité souhaitée et des cas d'utilisation.

Les codecs audio peuvent généralement contenir un canal audio (mono), deux canaux audio (stéréo) ou plusieurs canaux (par exemple, surround « 5.1 »). Par exemple, la voix humaine est enregistrée en utilisant un canal tandis que la musique utilise en général deux canaux ou plus. La qualité variera en fonction du débit, c'est-à-dire du nombre de bits utilisés par unité de temps de lecture.

Les codecs vidéo contiendront une séquence d'images, c'est-à-dire des images fixes et, pour les formats compressés, des mouvements entre ces images. La qualité variera en fonction du nombre d'images par seconde, de l'espace colorimétrique, de la résolution, etc.

Les formats de stockage multimédia contiendront des flux de codecs audio, des flux de codecs vidéo, des sous-titres et des méta-informations. Il les combine pour fournir l'audio ou la vidéo, avec des matériaux alternatifs ou améliorés. En général, une vidéo aura un flux de codec vidéo, un ou plusieurs flux de codec audio alternatifs, et peut avoir des sous-titres et des méta-informations.

4.2 A quoi servent l'audio et la vidéo ?

L'audio et la vidéo sont utilisés pour améliorer l'expérience avec les pages Web (par exemple, l'arrière-plan audio) pour diffuser de la musique, des vidéos familiales, des présentations, etc. ou la langue des signes.

4.3 Qu'est-ce que SMIL ?

SMIL est le langage d'intégration multimédia synchronisé, un langage basé sur XML pour décrire des présentations multimédia interactives. Il combine audio, vidéo, hypertexte, images dans le temps et dans l'espace, permettant des transitions visuelles entre les deux.

4.4 Qu'est-ce que le texte chronométré ?

Timed Text est un langage basé sur XML pour les médias de texte chronométré dans le but d'échanger entre les systèmes de création. Il peut également être utilisé directement comme format de diffusion, donc adapté au sous-titrage ou à la description vidéo.

4.5 Que sont les fragments de médias ?

Les fragments de média fournissent un moyen standard indépendant du format de média d'adresser des fragments de média, dans le temps et dans l'espace, sur le Web à l'aide d'identifiants (URL, URI, IRI).

4.6 Que sont les annotations multimédias ?

Les annotations multimédias fournissent des moyens de décrire les ressources multimédias, à l'aide d'un ensemble commun de propriétés. Ces annotations aident à transmettre des informations qui peuvent ensuite être réutilisées dans les moteurs de recherche ou les systèmes de balisage.

4.7 Exemples

Voici quelques exemples de SMIL, SVG et HTML respectivement avec du contenu vidéo :

```
<smil xmlns="http://www.w3.org/ns/SMIL">
  <body>
    <par>
      <video src="http://www.example.org/MyVideo" fill="freeze"/>
      <text src="http://www.example.org/MyCaption" fill="freeze" />
    </par>
  </body>
</smil>
```

```
<svg xmlns="http://www.w3.org/2000/svg"
     xmlns:xlink="http://www.w3.org/1999/xlink">
  <g>
    <video xlink:href="http://www.example.org/MyVideo"
           x="0" y="0" width="360" height="240" />
  </g>
</svg>
```

```
<!DOCTYPE html>
<html>
  <head>
```

(suite sur la page suivante)

(suite de la page précédente)

```
<title>My Video</title>
</head>
<body>
  <video src="http://www.example.org/MyVideo"
    width="360" height="240">
</body>
</html>
```

Pour plus d'informations consultez https://developer.mozilla.org/fr/docs/Web/SVG/SVG_animation_with_SMIL et <https://www.w3.org/TR/REC-smil/smil-animation.html>

Le Web est fondamentalement conçu pour fonctionner pour tous, quels que soient leur matériel, leurs logiciels, leur langue, leur emplacement ou leurs capacités. Lorsque le Web atteint cet objectif, il est accessible aux personnes ayant un large éventail de capacités auditives, motrices, visuelles et cognitives.

Ainsi, l'impact du handicap est radicalement modifié sur le Web car le Web supprime les barrières à la communication et à l'interaction auxquelles de nombreuses personnes sont confrontées dans le monde physique. Cependant, lorsque les sites Web, les applications, les technologies ou les outils sont mal conçus, ils peuvent créer des barrières qui empêchent les gens d'utiliser le Web.

L'accessibilité est essentielle pour les développeurs et les organisations qui souhaitent créer des sites Web et des outils Web de haute qualité, et ne pas empêcher les gens d'utiliser leurs produits et services.

La mission de la Web Accessibility Initiative (WAI) est de mener le Web à son plein potentiel pour être accessible, permettant aux personnes handicapées de participer de manière égale sur le Web.

5.1 Pourquoi : les arguments en faveur de l'accessibilité du Web

Le Web doit être accessible pour offrir un accès égal et des chances égales aux personnes ayant des capacités diverses. En effet, la Convention des Nations Unies relative aux droits des personnes handicapées reconnaît l'accès aux technologies de l'information et de la communication, y compris le Web, comme un droit humain fondamental.

L'accessibilité soutient l'inclusion sociale des personnes handicapées ainsi que d'autres personnes, telles que les personnes âgées, les habitants des zones rurales et les habitants des pays en développement.

L'accessibilité profite également aux personnes sans handicap. La vidéo Web Accessibility Perspectives montre des exemples de la façon dont l'accessibilité est essentielle pour les personnes handicapées et utile pour tout le monde dans diverses situations.

Il existe également une solide analyse de rentabilisation en faveur de l'accessibilité. L'accessibilité chevauche d'autres bonnes pratiques telles que la conception de sites Web mobiles, l'indépendance des appareils, l'interaction multimodale, la convivialité, la conception pour les utilisateurs plus âgés et l'optimisation des moteurs de recherche (SEO). Des études de cas montrent que les sites Web accessibles ont de meilleurs résultats de recherche, des coûts de maintenance réduits et une audience accrue, entre autres avantages. L'élaboration d'une analyse de rentabilisation de l'accessibilité du Web pour votre organisation détaille les avantages de l'accessibilité du Web.

5.2 Quoi : exemples d'accessibilité Web

Des sites Web et des outils bien conçus peuvent être utilisés par des personnes handicapées. Cependant, actuellement de nombreux sites et outils sont développés avec des barrières d'accessibilité qui rendent difficile voire impossible leur utilisation pour certaines personnes. Vous trouverez ci-dessous quelques exemples.

5.2.1 Texte alternatif pour les images

Les images doivent inclure un texte alternatif équivalent (texte alternatif) dans le balisage/le code.

Si le texte alternatif n'est pas fourni pour les images, les informations sur l'image sont inaccessibles, par exemple, aux personnes qui ne peuvent pas voir et utiliser un lecteur d'écran qui lit à haute voix les informations sur une page, y compris le texte alternatif de l'image visuelle.

Lorsqu'un texte alternatif équivalent est fourni, les informations sont disponibles pour les personnes aveugles, ainsi que pour les personnes qui désactivent les images (par exemple, dans les zones à bande passante chère ou faible). Il est également disponible pour les technologies qui ne peuvent pas voir les images, telles que les moteurs de recherche.

```
<img  
  alt= "Web accessibility initiative Logo"
```

5.2.2 La saisie au clavier

Certaines personnes ne peuvent pas utiliser de souris, y compris de nombreux utilisateurs plus âgés dont le contrôle de la motricité fine est limité. Un site Web accessible ne repose pas sur la souris ; il rend toutes les fonctionnalités disponibles à partir d'un clavier. Ensuite, les personnes handicapées peuvent utiliser des technologies d'assistance qui imitent le clavier, telles que la saisie vocale.

5.2.3 Transcriptions pour l'audio

Tout comme les images ne sont pas disponibles pour les personnes qui ne peuvent pas voir, les fichiers audio ne sont pas disponibles pour les personnes qui ne peuvent pas entendre. Fournir une transcription textuelle rend les informations audio accessibles aux personnes sourdes ou malentendantes, ainsi qu'aux moteurs de recherche et autres technologies qui ne peuvent pas entendre.

Il est facile et relativement peu coûteux pour les sites Web de fournir des transcriptions. Il existe également des services de transcription qui créent des transcriptions de texte au format HTML.

5.3 Comment : rendre votre site Web et vos outils Web accessibles

La plupart des bases de l'accessibilité sont assez faciles à mettre en œuvre. Cependant, si vous débutez dans le domaine de l'accessibilité, il faut du temps et des efforts pour apprendre les problèmes courants et les solutions. Voici les points de départ :

- [Principes d'accessibilité](#) — introduit les exigences d'accessibilité et les normes internationales.
- [Easy Checks - A First Review](#) — vous aide à commencer à tester l'accessibilité d'une page Web et à comprendre certains obstacles courants à l'accessibilité.
- [Conseils pour la mise en route](#) : fournit quelques considérations de base pour la conception, la rédaction et le développement.

Certains obstacles à l'accessibilité sont plus compliqués à éviter et les solutions demandent plus de temps et d'efforts de développement. W3C WAI fournit de nombreuses ressources pour vous aider, telles que des didacticiels et des supports liés à la présentation des WCAG 2.

L'utilisation d'outils de création prenant en charge l'accessibilité facilite la tâche des développeurs Web. Les navigateurs jouent également un rôle dans l'accessibilité. Ces rôles sont expliqués dans Composants essentiels de l'accessibilité Web.

L'accès au Web pour tous a été une préoccupation et un objectif fondamental du World Wide Web Consortium depuis le début. Malheureusement, il est facile de négliger les besoins des personnes de cultures différentes de la vôtre, ou qui utilisent des langues ou des systèmes d'écriture différents. Si vous le faites, vous créez des spécifications et du contenu qui présenteront des obstacles à l'utilisation de votre technologie ou de votre contenu pour de nombreuses personnes dans le monde.

6.1 Qu'est-ce que l'internationalisation ?

Si vous vous internationalisez, vous concevez ou développez votre contenu, votre application, vos spécifications, etc., d'une manière qui garantit qu'il fonctionnera bien ou qu'il pourra être facilement adapté pour des utilisateurs de toute culture, région ou langue.

Le mot « Internationalisation » est souvent abrégé en « i18n ». C'est une abréviation largement utilisée, dérivée du fait qu'il y a 18 lettres entre le “i” et le “n”.

6.2 Exemples

Un aspect fondamental de l'internationalisation est de s'assurer que la technologie prend en charge le texte dans n'importe quel système d'écriture du monde. C'est pourquoi les technologies du W3C reposent sur le jeu de caractères universel Unicode. Il peut être nécessaire de prendre également en charge d'autres jeux de caractères et encodages hérités.

e + ó ≡ é

Cependant, d'autres facteurs doivent être pris en compte lors de l'utilisation de caractères. Par exemple, les encodages basés sur Unicode permettent de stocker exactement le même texte en utilisant des combinaisons de caractères légèrement différentes. Pour plus d'efficacité et de précision dans la comparaison, le tri et l'analyse du texte, les différentes

séquences doivent être reconnues comme « canoniquement équivalentes ». Vous devez réfléchir à la manière de gérer cela lors du développement d'applications ou de spécifications qui exécutent ou reposent sur de telles tâches.

Parfois, différents systèmes d'écriture nécessitent un support spécial. Par exemple, le japonais, le chinois, le coréen et le mongol peuvent être écrits verticalement, de sorte que le W3C s'assure que CSS, SVG et XSL-FO permettront la prise en charge du texte vertical. Les méthodes d'alignement et de justification du texte sont également différentes pour de tels scripts, et différentes encore pour des scripts comme le thaï et le tibétain. D'autres conventions typographiques locales existent souvent pour des éléments tels que l'accentuation, les annotations, la numérotation des listes, etc. Ces approches typographiques doivent être supportées dans des feuilles de style.

L'arabe, l'hébreu, le persan, l'ourdou et les langues similaires mélangent du texte de droite à gauche et de gauche à droite sur la même ligne, et il est important de pouvoir contrôler la direction du contexte environnant pour que cela fonctionne correctement. Cela signifie que les développeurs de schémas et de formats doivent fournir aux auteurs des moyens de contrôler la direction de leur contenu. Les schémas, les langages de balisage et les formats doivent également prendre en charge un certain nombre d'autres constructions nécessaires pour une gestion efficace du contenu pendant la traduction et la localisation.

Image de texte bidirectionnel en arabe.

Si vous utilisez des formulaires HTML ou concevez des ontologies relatives aux noms et adresses de personnes, vous devrez réfléchir à la manière d'activer les nombreuses approches différentes de formatage des données qui sont possibles dans le monde. Vous devrez peut-être également prendre en charge des calendriers alternatifs, des fuseaux horaires et des heures d'été, des noms et des adresses sous forme native et translittérée, etc.

Les développeurs de contenu et les systèmes de gestion de contenu doivent également être préparés à faire face aux problèmes linguistiques et culturels. Par exemple, une phrase qui est construite en combinant plusieurs phrases ensemble dans une langue peut être impossible à traduire de manière sensée dans une langue avec une structure de phrase différente. Par exemple, dans la traduction japonaise de « Page 1 sur 34 », tous les éléments de la phrase seraient dans l'ordre inverse. Votre demande ne doit pas restreindre l'ordre dans lequel ces éléments peuvent être combinés. Les spécifications des technologies telles que les widgets et les navigateurs vocaux devraient également éviter d'enfermer les développeurs dans une syntaxe biaisée en anglais pour des choses telles que la composition de messages ou le déclenchement d'événements associés au texte.

Graphique illustrant l'ordre des mots entre l'anglais et l'hindi.

Graphique illustrant l'ordre des mots entre l'anglais et le finnois.

Un cercle peut représenter « oui » au Japon et une coche « non ».

Les problèmes culturels doivent également être pris en compte. Le symbolisme peut être spécifique à une culture. La coche signifie correct ou OK dans de nombreux pays. Dans certains pays, cependant, comme le Japon, cela peut être utilisé pour signifier que quelque chose est incorrect. Les localisateurs japonais peuvent avoir besoin de convertir les coches en cercles (leur symbole pour « correct ») dans le cadre du processus de localisation.

Ce ne sont que quelques exemples parmi tant d'autres. Le message clé est que la conception (qu'il s'agisse d'un langage de balisage, d'un protocole, d'un système de gestion de contenu, d'un widget ou d'une application, etc.) doit

être suffisamment flexible pour s'adapter aux besoins locaux.

7.1 Qu'est-ce que PHP ?

- PHP est l'acronyme de « PHP : Hypertext Preprocessor »
- PHP est un langage de script open source largement utilisé
- Les scripts PHP sont exécutés sur le serveur
- PHP est gratuit à télécharger et à utiliser

7.2 Qu'est-ce qu'un fichier PHP ?

Les fichiers PHP peuvent contenir du texte, du HTML, du CSS, du JavaScript et du code PHP. Le code PHP est exécuté sur le serveur et le résultat est renvoyé au navigateur en tant que code HTML brut. Les fichiers PHP ont l'extension *.php*

7.3 Que peut faire PHP ?

- PHP peut générer du contenu de page dynamique
- PHP peut créer, ouvrir, lire, écrire, supprimer et fermer des fichiers sur le serveur
- PHP peut collecter des données de formulaire
- PHP peut envoyer et recevoir des cookies
- PHP peut ajouter, supprimer, modifier des données dans votre base de données
- PHP peut être utilisé pour contrôler l'accès des utilisateurs
- PHP peut crypter les données

Avec PHP, vous n'êtes pas limité à la sortie HTML. Vous pouvez produire des images, des fichiers PDF et même des films Flash. Vous pouvez également sortir n'importe quel texte, tel que XHTML et XML.

7.4 Pourquoi PHP ?

- PHP fonctionne sur diverses plateformes (Windows, Linux, Unix, Mac OS X, etc.)
- PHP est compatible avec presque tous les serveurs utilisés aujourd'hui (Apache, IIS, etc.)
- PHP prend en charge un large éventail de bases de données
- PHP est gratuit. Téléchargez-le à partir de la ressource PHP officielle : www.php.net
- PHP est facile à apprendre et fonctionne efficacement côté serveur

7.5 Quoi de neuf en PHP 7

- PHP 7 est beaucoup plus rapide que la précédente version stable populaire (PHP 5.6)
- PHP 7 a amélioré la gestion des erreurs
- PHP 7 prend en charge des déclarations de type plus strictes pour les arguments de fonction
- PHP 7 prend en charge de nouveaux opérateurs (comme l'opérateur de vaisseau spatial : `<=>`)

7.6 Installation de PHP

7.6.1 De quoi ai-je besoin ?

Pour commencer à utiliser PHP, vous pouvez :

- Trouver un hébergeur avec support PHP et MySQL
- Installez un serveur Web sur votre propre PC, puis installez PHP et MySQL

7.6.2 Utiliser un hébergeur avec support PHP

Si votre serveur a activé la prise en charge de PHP, vous n'avez rien à faire. Créez simplement des fichiers `.php`, placez-les dans votre répertoire Web et le serveur les analysera automatiquement pour vous. Vous n'avez pas besoin de compiler quoi que ce soit ou d'installer des outils supplémentaires. Parce que PHP est gratuit, la plupart des hébergeurs proposent un support PHP.

7.6.3 Configurer PHP sur votre propre PC

Cependant, si votre serveur ne supporte pas PHP, vous devez :

- installer un serveur web
- installer PHP
- installer une base de données, telle que MySQL

Le site Web officiel de PHP (PHP.net) contient des instructions d'installation pour PHP : <http://php.net/manual/en/install.php> Compileur/éditeur PHP en ligne

Syntaxe de PHP

Un script PHP est exécuté sur le serveur et le résultat HTML brut est renvoyé au navigateur.

8.1 Syntaxe PHP de base

Un script PHP peut être placé n'importe où dans le document.

Un script PHP commence par `< ?php` et se termine par `?>` :

```
<?php
    // Le code PHP va ici
?>
```

L'extension de fichier par défaut pour les fichiers PHP est `.php`.

Un fichier PHP contient normalement des balises HTML et du code de script PHP. Ci-dessous, nous avons un exemple de fichier PHP simple, avec un script PHP qui utilise une fonction PHP intégrée « `echo` » pour afficher le texte « Hello World ! » sur une page Web :

```
<!DOCTYPE html>
<html>
  <body>
    <h1>Ma première page PHP</h1>
    <?php
      echo "Bonjour tout le monde !";
    ?>

  </body>
```

Note : les instructions PHP se terminent par un point-virgule (;).

8.1.1 Sensibilité à la casse PHP

En PHP, les mots-clés (par exemple, if, else, while, echo, etc.), les classes, les fonctions et les fonctions définies par l'utilisateur ne sont pas sensibles à la casse.

Dans l'exemple ci-dessous, les trois instructions echo ci-dessous sont égales et légales :

```
<!DOCTYPE html>
<html>
  <body>

    <?php
      ECHO "Bonjour tout le monde !<br>" ;
      echo "Bonjour tout le monde !<br>" ;
      EcHo "Bonjour tout le monde !<br>" ;

    ?>
  </body>
</html>
```

Note : Cependant ; tous les noms de variables sont sensibles à la casse !

Regardez l'exemple ci-dessous ; seule la première instruction affichera la valeur de la variable \$color ! En effet, \$color, \$COLOR et \$coLOR sont traités comme trois variables différentes :

```
<!DOCTYPE html>
<html>
  <body>
    <?php
      $color = "rouge" ;
      echo "Ma voiture est". $color. "<br>" ;
      echo "Ma maison est " . $COLOR. "<br>" ;
      echo "Mon bateau est " . $coLOR. "<br>" ;

    ?>
  </body>
</html>
```

8.1.2 Exercices PHP

Insérez la partie manquante du code ci-dessous pour afficher « Bonjour le Monde ».

```
"Bonjour le monde";
```

8.2 Commentaires en PHP

Un commentaire dans le code PHP est une ligne qui n'est pas exécutée dans le cadre du programme. Son seul but est d'être lu par quelqu'un qui regarde le code.

Les commentaires peuvent être utilisés pour :

- Laisser les autres comprendre votre code
- Vous rappeler ce que vous avez fait - La plupart des programmeurs ont eu l'habitude de revenir à leur propre travail un an ou deux plus tard et d'avoir à repenser ce qu'ils ont fait. Les commentaires peuvent vous rappeler ce que vous pensiez lorsque vous avez écrit le code

PHP prend en charge plusieurs manières de commenter.

Syntaxe pour les commentaires sur une seule ligne :

```
<!DOCTYPE html>
<html>
  <body>
    <?php
      // Ceci est un commentaire sur une seule ligne
      # Ceci est également un commentaire sur une seule ligne
    ?>
  </body>
</html>
```

Syntaxe pour les commentaires sur plusieurs lignes :

```
<!DOCTYPE html>
<html>
  <body>
    <?php
      /*
        Ceci est un bloc de commentaires de plusieurs lignes
        qui s'étend sur plusieurs
        lignes
      */
    ?>
  </body>
</html>
```

Utiliser des commentaires pour omettre des parties du code :

```
<!DOCTYPE html>
<html>
  <body>
    <?php
      // Vous pouvez également utiliser des commentaires pour omettre des parties d
      ↪ 'une ligne de code
      $x = 5 /* + 15 */ + 5;
      echo $x;
    ?>
  </body>
</html>
```

Pour plus d'information visitez https://fr.wikibooks.org/wiki/Programmation_PHP

8.3 Les formulaires en PHP

Les superglobales PHP `$_GET` et `$_POST` sont utilisées pour collecter des données de formulaire.

8.3.1 PHP - Un formulaire HTML simple

L'exemple ci-dessous affiche un formulaire HTML simple avec deux champs de saisie et un bouton d'envoi :

```
<!DOCTYPE html>
<html>
<body>
  <form action="bienvenue.php" method="post">
    Nom : <input type="text" name="name"><br>
    Courriel : <input type="text" name="email"><br>
    <input type="submit">
  </form>
</body>
</html>
```

Lorsque l'utilisateur remplit le formulaire ci-dessus et clique sur le bouton *Soumettre*, les données du formulaire sont envoyées pour traitement dans un fichier PHP nommé « *bienvenue.php* ». Les données du formulaire sont envoyées avec la méthode *HTTP POST*.

Pour afficher les données soumises, vous pouvez simplement faire *echo* à toutes les variables. Le fichier « *bienvenue.php* » ressemble à ceci :

```
<!DOCTYPE html>
<html>
<body>
  Bienvenue <?php echo $_POST["name"]; ?><br>
  Votre adresse e-mail est : <?php echo $_POST["email"] ; ?>
</body>
</html>
```

La sortie pourrait être quelque chose comme ceci :

Note : Bienvenue Jean Votre adresse e-mail est `john.doe@example.com`

Le même résultat pourrait également être obtenu en utilisant la méthode *HTTP GET* :

```
<html>
<body>
  <form action="welcome_get.php" method="get">
    Nom : <input type="text" name="name"><br>
    Courriel : <input type="text" name="email"><br>
    <input type="submit">
  </form>
</body>
</html>
```

et « *welcome_get.php* » ressemble à ceci :

```
<html>
<body>
```

(suite sur la page suivante)

(suite de la page précédente)

```
Bienvenue <?php echo $_GET["name"]; ?><br>
Votre adresse e-mail est : <?php echo $_GET["email"] ; ?>
</body>
</html>
```

Le code ci-dessus est assez simple. Cependant, il manque le plus important. Vous devez valider les données du formulaire pour protéger votre script des codes malveillants.

Pensez SÉCURITÉ lors du traitement des formulaires PHP !

Cette page ne contient aucune validation de formulaire, elle montre simplement comment vous pouvez envoyer et récupérer des données de formulaire.

Cependant, les pages suivantes montreront comment traiter les formulaires PHP en gardant à l'esprit la sécurité ! Une validation correcte des données du formulaire est importante pour protéger votre formulaire des pirates et des spammeurs !

8.3.2 GET vs POST

GET et POST créent tous deux un tableau (par exemple, `array(key1 => value1, key2 => value2, key3 => value3, ...)`). Ce tableau contient des paires clé/valeur, où les clés sont les noms des contrôles de formulaire et les valeurs sont les données d'entrée de l'utilisateur.

GET et POST sont tous deux traités comme `$_GET` et `$_POST`. Ce sont des superglobales, ce qui signifie qu'elles sont toujours accessibles, quelle que soit la portée - et vous pouvez y accéder à partir de n'importe quelle fonction, classe ou fichier sans avoir à faire quoi que ce soit de spécial.

`$_GET` est un tableau de variables passé au script courant via les paramètres d'URL.

`$_POST` est un tableau de variables passé au script courant via la méthode HTTP POST.

8.3.3 Quand utiliser GET ?

Les informations envoyées depuis un formulaire avec la méthode GET sont visibles par tous (tous les noms et valeurs des variables sont affichés dans l'URL). GET a également des limites sur la quantité d'informations à envoyer. La limitation est d'environ 2000 caractères. Cependant, comme les variables sont affichées dans l'URL, il est possible de mettre la page en signet. Cela peut être utile dans certains cas.

GET peut être utilisé pour envoyer des données non sensibles.

Note : GET ne doit JAMAIS être utilisé pour envoyer des mots de passe ou d'autres informations sensibles !

8.3.4 Quand utiliser POST ?

Les informations envoyées à partir d'un formulaire avec la méthode POST sont invisibles pour les autres (tous les noms/valeurs sont intégrés dans le corps de la requête HTTP) et n'ont aucune limite sur la quantité d'informations à envoyer.

De plus, POST prend en charge des fonctionnalités avancées telles que la prise en charge de l'entrée binaire en plusieurs parties lors du téléchargement de fichiers sur le serveur.

Cependant, comme les variables ne sont pas affichées dans l'URL, il n'est pas possible de mettre la page en signet.

Les développeurs préfèrent POST pour envoyer des données de formulaire.

Ensuite, voyons comment nous pouvons traiter les formulaires PHP de manière sécurisée !

8.3.5 Exercices PHP

Si le formulaire suivant est soumis, comment pouvez-vous, dans welcome.php, afficher la valeur du champ « fname » ?

```
<form action="welcome.php" method="get">
  Prénom : <input type="text" name="fname">
</form>

<html>
  <body>
    Bienvenue <?php echo _____ ; ?>
  </body>
</html>
```


CHAPITRE 9

Indices and tables

- `genindex`
- `modindex`
- `search`