Check-list pour l'accessibilité mobile

Ce document fournit une liste concise des points à vérifier par les développeuses et développeurs pour garantir l'accessibilité d'une application mobile. Ce document est amené à évoluer pour tenir compte de nouvelles bonnes pratiques.

[La couleur](https://developer.mozilla.org/fr/docs/Web/Accessibility/Mobile_accessibility_checklist)

* Le contraste des couleurs **DOIT** être conforme aux [exigences du niveau AA du WCAG 2.1](https://www.w3.org/TR/WCAG/) :
  + Un contraste dont le ratio est de 4.5:1 pour les textes normaux (dont la fonte est inférieure à 18 points ou 14 points en gras) ;
  + Un contraste dont le ratio est de 3:1 pour les grands textes (18 points minimum ou 14 points en gras).
* L'information transmise par la couleur **DOIT** également être disponible par d'autres moyens (texte souligné pour les liens, etc.).

[La visibilité](https://developer.mozilla.org/fr/docs/Web/Accessibility/Mobile_accessibility_checklist)

* Les techniques de masquage du contenu, telles que l'opacité nulle, l'ordre d'indexation en « z » et le placement hors écran, **NE DOIVENT PAS** être utilisées exclusivement pour gérer la visibilité.
* Tout ce qui est autre, que l'écran actuellement visible, **DOIT** être *vraiment* invisible (particulièrement pertinent pour les apps à page unique avec plusieurs « *cartes* ») :
  + Utilisez l'attribut hidden ou les propriétés de style visibility ou display.
  + Sauf si cela est absolument inévitable, l'attribut aria-hidden**NE DOIT PAS** être utilisé.

[Le focus](https://developer.mozilla.org/fr/docs/Web/Accessibility/Mobile_accessibility_checklist)

* Tous les éléments activables **DOIVENT** être focusables :
  + Les contrôles standard tels que les liens, les boutons et les champs de formulaire sont accessibles par défaut.
  + Les contrôles non standard **DOIVENT** avoir un [rôle ARIA (en-US)](https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/Roles) approprié qui leur est attribué, comme button, link ou checkbox.
* Le focus **DOIT** être traité dans un ordre logique et de manière cohérente.

[Les équivalents textuels](https://developer.mozilla.org/fr/docs/Web/Accessibility/Mobile_accessibility_checklist)

* Un équivalent textuel **DOIT** être fourni pour chaque élément non textuel non strictement présenté dans l'application.
  + Utilisez *alt* et *title* lorsque cela est approprié (voir l'article de Steve Faulkner sur l'[Utilisation de l'attribut HTML title](https://www.tpgi.com/using-the-html-title-attribute-updated/)).
  + Si les attributs ci-dessus ne sont pas applicables, utilisez les [États et propriétés ARIA](https://www.w3.org/TR/wai-aria-1.1/) appropriés tels que aria-label, aria-labelledby, ou aria-describedby.
* Les images de texte **DOIVENT** être évitées.
* Tous les composants de l'interface utilisateur ayant un texte visible (ou une image de texte) comme étiquette **DOIVENT** avoir le même texte disponible dans le [nom](https://www.w3.org/TR/WCAG21/) programmatique du composant. [WCAG 2.1 : Étiquette dans le nom.](https://www.w3.org/WAI/WCAG21/Understanding/label-in-name.html)
* Tous les contrôles de formulaire **DOIVENT** avoir des étiquettes (éléments [<label>](https://developer.mozilla.org/fr/docs/Web/HTML/Element/Label)) pour le bénéfice des utilisateurs de lecteurs d'écran.

[La gestion des états](https://developer.mozilla.org/fr/docs/Web/Accessibility/Mobile_accessibility_checklist)

Les contrôles standard tels que les boutons radio et les cases à cocher sont gérés par le système d'exploitation. Cependant, pour d'autres contrôles personnalisés, les changements d'état **DOIVENT** être fournis via [les états ARIA](https://www.w3.org/TR/wai-aria-1.1/) tels que aria-checked, aria-disabled, aria-selected, aria-expanded et aria-pressed.

[L'orientation](https://developer.mozilla.org/fr/docs/Web/Accessibility/Mobile_accessibility_checklist)

* Le contenu **NE DOIT PAS** être limité à une seule orientation, comme le portrait ou le paysage, sauf si cela est essentiel. [WCAG 2.1 : Orientation](https://www.w3.org/WAI/WCAG21/Understanding/orientation.html)
  + Des exemples de cas où une orientation est essentielle sont une application pour un piano ou un chèque de banque.

[Directives générales](https://developer.mozilla.org/fr/docs/Web/Accessibility/Mobile_accessibility_checklist)

* Un titre d'application **DOIT** être fourni.
* Les titres **NE DOIVENT PAS** rompre la structure hiérarchique

<h1>Titre de premier niveau</h1> <h2>Titre secondaire</h2> <h2>Un autre titre secondaire</h2> <h3>Titre de bas niveau</h3>

* L'[ARIA Landmark Roles](https://www.washington.edu/accessibility/web/landmarks/)**DOIT** être utilisé pour décrire une structure d'application ou de document, telle que banner, complementary, contentinfo, main, navigation, search.
* Pour les événements tactiles, au moins un des éléments suivants **DOIT** être vrai ([WCAG 2.1 : Annulation du pointeur](https://www.w3.org/WAI/WCAG21/Understanding/pointer-cancellation.html)) :
  + L'événement de descente **NE DOIT PAS** être utilisé pour déclencher une action.
  + L'action est déclenchée par l'événement « up » et une option permettant d'interrompre l'action avant son achèvement est disponible ou une option permettant d'annuler l'action après son achèvement.
  + L'événement de montée annulera toute action déclenchée par un événement de descente.
  + Il est essentiel de déclencher l'action sur l'événement de descente. Par exemple, pour jouer à un jeu ou à une application de piano.
* Les cibles tactiles **DOIVENT** être suffisamment grandes pour que l'utilisateur puisse interagir avec elles (voir [BBC Mobile Accessibility Guidelines](https://www.bbc.co.uk/guidelines/futuremedia/accessibility/mobile/design/touch-target-size) pour des directives utiles sur la taille des cibles tactiles).