

L'accessibilité des formulaires Web



Pour la présentation, seront utilisés :

Firefox

NVDA

Les fichiers d'exemples se trouvent à a11yqc.org/atelier_forms/base.html

Note : Le code JavaScript présent dans ces exemples n'a été codé que pour démontrer certains comportements propre au contenu présenté dans le cadre de cette atelier. Il n'a pas été codé pour être robuste et fonctionner dans toutes les situations.

Outils de validation

➤ Extensions Firefox

- Firebug
- Web Developer Toolbar
- HeadingsMap
- WCAG Contrast Checker
- WAVE

➤ NVDA (Visionneuse de parole)

➤ Navigation avec le clavier

Mode formulaire et mode document

Mode formulaire

- N'est pas le mode par défaut
- Raccourcis clavier NVDA désactivés afin de permettre la saisie de données dans les champs

Mode document

- Mode par défaut
- Raccourcis clavier NVDA activés

Navigation au clavier

- NVDA passe en mode formulaire si focus donné sur un champ qui le requiert
- NVDA passe en mode document si focus donné sur un champ qui ne requiert pas le mode formulaire
- Touches
 - Tabulation/Shift + Tabulation pour naviguer
 - Espace ou Entrée pour passer en mode formulaire
 - Échap pour passer en mode document
 - Forcer l'entrée ou la sortie du mode formulaire (NVDA + Espace)

Étiquette et champ

➤ Lier le champ et l'étiquette

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

➤ Utiliser l'attribut *title* **SI** manque d'espace

```
<input id="nom" title="Nom" type="text" />
```

➤ Utiliser l'attribut *aria-label* **SI** manque d'espace

```
<input id="nom" aria-label="Nom"  
type="text" />
```

Groupement de champs

- Boutons radio
- Groupe de cases à cocher
- Champs liés
 - Champs pour une date (année, mois, jour)
 - Informations liées (ex : adresse postale)

```
<fieldset>  
  <legend>Date</legend>  
  <input id="annee" title="Année" type="text" />  
  <input id="mois" title="Mois" type="text" />  
  <input id="jour" title="Jour" type="text" />  
</fieldset>
```

Champ désactivé ou en lecture seule

Champ désactivé

- Navigation au clavier avec tabulation passe par-dessus
- Lecteur d'écran ne communique pas l'existence de ce champ

```
<input type="text" disabled="disabled" />
```

Champ en lecture seule

- Navigation au clavier avec tabulation reçoit le focus
- Lecteur d'écran communique la notion de lecture seule

```
<input type="text" readonly="readonly" />
```

Information entre les champs

- Se rappeler que l'utilisateur navigue par tabulation
- L'utilisateur passe donc par-dessus le texte entre les champs
- L'information peut être liée avec *aria-describedby*

Un champ qui est l'étiquette d'un autre champ

- Il est possible que, selon le contexte, un champ fasse office d'étiquette pour un autre champ
- Ajouter un attribut *title* sur le champ sans étiquette `<label>`
- Lier le champ sans étiquette `<label>` au champ lui servant d'étiquette avec *aria-labelledby*

Gestion des erreurs

- Liste d'erreurs au début du formulaire
- Liste d'erreurs comprend des liens qui mènent vers les champs en erreurs
- Focus est donné à la liste d'erreurs au clic du bouton **Envoyer**
- Message d'erreur de chaque champ est lié par aria-describedby
- Mot **Erreur** en texte hors écran placé au début du `</label>`

Certaines problématiques

Champ ajouté dynamiquement

➤ Problématique

- Au *focus out*, le focus sera donné au champ qui se situe après celui ajouté dynamiquement

Champ ajouté dynamiquement

➤ Piste de solution

- Créer un garde non sémantique qui est le gardien du focus
- Garde reçoit le focus et le redonne au champ ajouté dynamiquement

Naviguer rapidement dans un long formulaire

➤ Problématique

- Pas de raccourcis clavier pour naviguer de `<legend>` en `<legend>`
- L'en-tête de section n'a pas de notion de groupage qui est lue

Naviguer rapidement dans un long formulaire

➤ Piste de solution

- Garder le `<fieldset>`
- Remplacer le `<legend>` par un en-tête de section (ex : `<h2>`)
- Attribut `aria-labelledby` sur le `<fieldset>` qui pointe vers l'en-tête de section

Zone de texte avec un maximum de caractères

➤ Problématique

- Attribut *maxlength* ne fournit aucune rétroaction
- Si un message était redit à chaque touche, cela deviendrait une cacophonie
- Lorsqu'une touche est appuyée, la touche coupe le message codé avec l'attribut *aria-live*
- On perd les caractères en trop sans le savoir en copiant collant du texte et c'est considéré comme valide à l'envoi du formulaire

Zone de texte avec un maximum de caractères

➤ Piste de solution

- Permettre que l'utilisateur puisse copier-coller plus de caractères
- Ajouter un délai lorsqu'aucune touche n'est appuyé pour annoncer ce qu'il reste comme nombre de caractères disponibles

Éléments pouvant poser un problème pour l'accessibilité

Cases à cocher

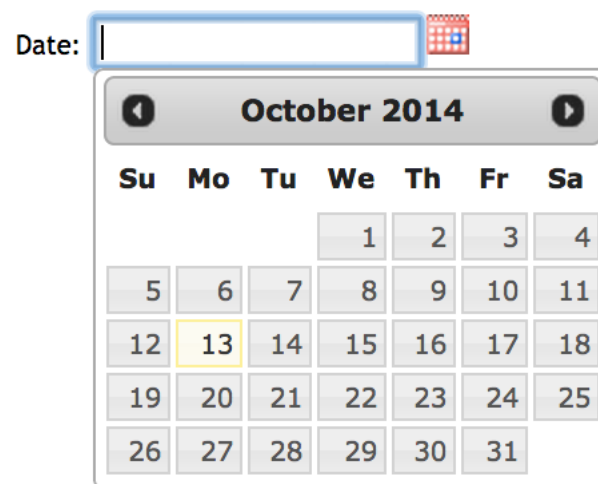
- Problème lorsqu'un élément fait sortir le mode formulaire avant ou après un groupe de cases à cocher
- Les cases à cocher sortent du mode formulaire
- NVDA ne mentionne alors pas le groupage (ni le *<fieldset>* ni le *<legend>*)

Datepicker

➤ Plusieurs non accessibles sur le marché

➤ À surveiller

- Au clic de l'icône de calendrier, le focus ne se donne pas au calendrier
- L'ordre de lecture n'est pas respecté dans le calendrier
- Très difficile de naviguer dans le calendrier
- Des balises non sémantiques sont utilisées dans le calendrier (ex : `<div>`)



Datepicker

➤ Si *Datepicker* non accessible

- Avoir un champ texte comme alternative qui peut se remplir manuellement
- S'assurer que l'icône de calendrier ne puisse avoir le focus
- S'assurer que l'icône de calendrier puisse être ignorée par les lecteurs d'écran (attribut *aria-hidden*)

Attributs *required* et *aria-required*

required

- NVDA lit « obligatoire »
- Premier focus dans un champ texte vide, NVDA lit « saisie incorrecte »
- Si le *<fieldset>* contient au moins un champ avec l'attribut *required*, NVDA lit « saisie incorrecte » pour tout le groupage

aria-required

- NVDA lit « obligatoire »
- NVDA ne mentionne pas « saisie incorrecte »
- Si le *<fieldset>* possède l'attribut *aria-required*, le groupe sera annoncé comme obligatoire

Masque

- Problème similaire avec l'attribut *maxlength* : aucune rétroaction
- Rien n'est communiqué si l'utilisateur saisie un mauvais caractère