

Développement front

MMI 2 – TP#5 S3



Modale

- Désigne un élément qui prend la main sur l'ensemble de la page
 - Action impossible hors de la modale (inert)
- Une fois affiché, on ne peut pas interagir avec le reste de la page
- Équivalent d'une alert sur smartphone ou en javascript

Point technique – Attribut “inert”

- Rend un élément (et ses enfants) non-interagissable
- Fonctionne avec toutes les balises
 - Contrairement à l'attribut disabled
- Activé par défaut sur la page quand un `<dialog>` est ouvert en mode modale

Modale

- Apparaît au-dessus de tout sur le site
- Contrevient aux limites de la modale de l'ancien temps :
 - Impossibilité de faire prendre la main
 - Ne crée pas un focus trap

Définition – Focus trap

"Fait de restreindre le focus clavier dans un contexte spécifique dans le but de forcer une action utilisateur qui est en navigation clavier"

Balise <dialog>

- Répond aux caractéristiques de la modale
- Apparaît avec un fond en arrière-plan
 - Uniquement si affiché en javascript
- Deux modes : modal et non-modal
- Plusieurs dialog peuvent être ouverts en même temps

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Elements/dialog>

Balise <dialog>



```
<dialog open>  
  <p>Je suis une &lt;dialog> qui est une non-modal</p>  
</dialog>
```

Apparaît au chargement de la page, mais sans arrière-plan



```
<dialog>  
  <p>Je suis une &lt;dialog> qui est une modal</p>  
</dialog>
```

Apparaîtra avec du javascript et un arrière-plan (ou non)

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Elements/dialog>

Balise <dialog> - Note

- C'est la responsabilité du navigateur de gérer l'ouverture / fermeture du dialog
 - Si vous appliquez “display : flex” (ex), il sera impossible de fermer / ouvrir la modale

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Elements/dialog>

Balise <dialog> - Note



```
<dialog class="flex">
  <!-- [ ... ] -->
  <button type="button" data-fermeture-modal>Fermer</button>
</dialog>
```

A cause du “display: flex”, il est impossible de fermer la modale

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Elements/dialog>

Balise <dialog>



JS

```
const dialog = document.querySelector("dialog");  
dialog.showModal();
```

Notre dialog s'affiche **avec** un arrière-plan (backdrop) grâce au javascript

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Elements/dialog>

Balise <dialog>



JS

```
const dialog = document.querySelector("dialog");  
dialog.show();
```

Notre dialog s'affiche **sans** un arrière-plan (backdrop) grâce au javascript

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Elements/dialog>

Pseudo-élément ::backdrop

- Représente l'arrière-plan d'un élément modal **uniquement**
- Accepte les mêmes valeurs que la propriété CSS "background"
- Fonctionne également avec les éléments en plein écran (API Fullscreen)

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/CSS/::backdrop>

Balise <dialog> - Fermeture

- Géré par l'attribut "closedby"
 - none (défaut), closedrequest, any
- Trois possibilités :
 - Touche Echap (any, closedrequest)
 - Clic à l'extérieur du <dialog> (any)
 - Déclencheur manuel. Ex : bouton (none)

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Elements/dialog>
- <https://developer.mozilla.org/en-US/docs/Web/HTML/Reference/Elements/dialog#closedby>

Balise <dialog> - Fermeture



JS

```
const button = document.querySelector("button");  
const dialog = document.querySelector("dialog");  
  
button.addEventListener("click", (e) => {  
    dialog.close();  
});
```

Au clic sur le bouton, <dialog> sera fermé (mode “modale” ou non)

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Elements/dialog>

Balise <dialog> - Fermeture



```
<dialog>
  <!-- [ ... ] -->
  <form method="dialog">
    <button type="submit">
      Fermer
    </button>
  </form>
</dialog>
```

Alternative, sans javascript, on joue sur la méthode “dialog” du formulaire

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Elements/dialog>

Pratiquons ! - Découvrons les modales et non modales (Partie 1)

Pré-requis :

- Avoir la ressource *ressources/modal-or-not-modal*

A télécharger ici :

https://github.com/DanYellow/cours/raw/refs/heads/main/s3-integration-web/travaux-pratiques/numero-5/s3-integration-web_travaux-pratiques_numero-5.ressources.zip

Attribut popover

- Permet d'ouvrir une balise au-dessus de la page
 - Fonctionne avec toutes la balises
- Ne requiert pas de javascript pour s'ouvrir
- Possibilité d'avoir un arrière-plan

Source(s) :

- https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Global_attributes/popover

Attribut popover

- Ouverture exclusive : deux popovers ne peuvent être ouverts en même temps
 - S'ils sont en mode “auto” ou “hint”
- Peut être placé n'importe où sur la page

Source(s) :

- https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Global_attributes/popover

Attribut popover

- Trois valeurs possibles :
 - auto (défaut) : Se ferme quand on clique à l'extérieur ou touche "echap"
 - manual : Nécessite une action manuelle pour être fermé
 - hint (expérimental) : Apparaît / disparaît au survol

Source(s) :

- https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Global_attributes/popover

Attribut popover



JS

```
<button popovertarget="mypopover">Afficher popover</button>  
<div id="mypopover" popover>Contenu du popover</div>
```

Grâce à l'attribut "id" et "popovertarget", on lie deux éléments
(Il ne faut pas oublier l'attribut "popover")

L'attribut "popovertarget" ne fonctionne qu'avec la balise <button>

Source(s) :

- https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Global_attributes/popover

Attribut popover - Ouverture

- Trois possibilités :
 - Clic (utilisation des attributs)
 - Javascript (ensemble de méthodes)
 - Survol (popover="hint")
 - Ne fonctionne que sur Chrome (10/25) partiellement

Source(s) :

- https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Global_attributes/popover
- <https://una.im/popover-hint/> - anglais

Attribut popover - Fermeture

- Quatre possibilités :
 - Touche Echap
 - Clic à l'extérieur du <dialog>
 - Déclencheur manuel. Ex : bouton
 - Ouverture d'un nouveau popover

Source(s) :

- https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Global_attributes/popover

Pratiquons ! - Découvrons les modales et non modales (Partie 2)

Pré-requis :

- Avoir la ressource *ressources/modal-or-not-modal*

A télécharger ici :

https://github.com/DanYellow/cours/raw/refs/heads/main/s3-integration-web/travaux-pratiques/numero-5/s3-integration-web_travaux-pratiques_numero-5.ressources.zip

Attribut popovertargetaction

- Permet de définir l'action à effectuer sur un popover quand on clic sur un bouton
- Trois valeurs possibles :
 - hide : cache le popover
 - show : affiche le popover
 - toggle (défaut) : affiche/cache le popover

Source(s) :

- <https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Elements/button#popovertargetaction>

Placement du popover

- Placement libre sur la page
 - Peut être fastidieux pour certains cas. Ex : bulle d'aide pour un formulaire
- Futur : utilisation de l'API "anchor-positioning"
 - Pas géré par Firefox (10/25)

Source(s) :

- https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_anchor_positioning
- <https://css-irl.info/anchor-positioning-and-the-popover-api/> - anglais - exemple

Transition

- Utilisation de la règle-at @starting-style
- Fonctionne à l'ouverture et fermeture
- Utilisation de l'attribut [open] et de la pseudo-classe :popover-open
 - <dialog> et popover respectivement

Source(s) :

- <https://nerdy.dev/using-starting-style-and-transition-behavior-for-enter-and-exit-stage-effects#dialog-and-popover>

Transition



```
[popover]:popover-open {  
  transition: opacity 3s;  
  opacity: 1;  
  
  @starting-style {  
    opacity: 0;  
  }  
}
```

Le popover fera une transition d'opacité lors de son ouverture

Source(s) :

- <https://nerdy.dev/using-starting-style-and-transition-behavior-for-enter-and-exit-stage-effects#dialog-and-popover>

Transition



```
[popover] {  
  transition: opacity 0.5s, display 0.5s allow-discrete;  
  opacity: 0;  
}
```

Le popover fera une transition d'opacité lors de sa fermeture
(ne fonctionne pas sur tous les navigateurs)

Source(s) :

- <https://nerdy.dev/using-starting-style-and-transition-behavior-for-enter-and-exit-stage-effects#dialog-and-popover>

Transition

- Possibilité de faire une transition sur le `::backdrop` (arrière-plan)

Source(s) :

- <https://nerdy.dev/using-starting-style-and-transition-behavior-for-enter-and-exit-stage-effects#dialog-and-popover>

Pratiquons ! - Découvrons les modales et non modales (Partie 3)

Pré-requis :

- Avoir la ressource *ressources/modal-or-not-modal*

A télécharger ici :

https://github.com/DanYellow/cours/raw/refs/heads/main/s3-integration-web/travaux-pratiques/numero-5/s3-integration-web_travaux-pratiques_numero-5.ressources.zip

Règle empirique : modale ou popover ?

- L'utilisateur peut ignorer l'information :
 - Popover
- On veut mettre un arrière-plan :
 - Modale ou popover
- L'utilisateur ne peut pas ignorer l'information :
 - Modale
- Le focus clavier doit être restreint à une zone :
 - Modale

Résumé : modale ou popover

	popover	<dialog> (modal)	<dialog> (non modal)
Arrière-plan	oui	oui	non
Conforme à la norme “modale”	non	oui	non
Affiché au-dessus de tout	oui	oui	oui
Fermeture (Change en fonction du paramétrage)	Touche echap Clic extérieur Action utilisateur Touche echap Clic extérieur Action utilisateur Touche echap Clic extérieur Action utilisateur Touche echap Clic extérieur		

Questions ?

