

Pronote : création d'une page d'options pour l'extension

Principe

L'objectif est simple, ajouter une page d'options afin de personnaliser ou de compléter l'extension.

On va utiliser deux fichiers, `pageOptions.html`, qui va contenir la partie graphique, et `options.js`, qui permettra l'interaction entre l'utilisateur et l'extension.

On se contentera d'une page qui propose à l'utilisateur de saisir le lien menant vers le pronote de son établissement, avant de le rediriger vers ce lien une fois qu'il aura cliqué sur un bouton `<continuer>`.

Création de la page

Créons tout d'abord les deux fichiers, `pageOptions.html`, et `options.js`. Il est nécessaire d'inclure la page d'options dans le manifest :

```
{
  "name" : "Premiere_Extension",
  "manifest_version": 2,
  "version" : "1.0",
  "description" : "Test1",
  **"options_page": "pageOptions.html",**
  "background_page": "background.html",

  "browser_action" :
  {
    "default_icon" : "icone.png",
    "default_popup" : "affttexte.html"
  },

  "content_scripts": [
  {
    "matches": ["https://0870056n.index-education.net/pronote/eleve.html"],
    "js": ["commSite.js"]
  }
  ]
}
```

Nous commencerons par coder la partie html. Il faut tout simplement ajouter un texte, une textbox et un bouton continuer. Pour que la page puisse communiquer avec le fichier `pageOptions.js`, il faut l'inclure. Ci-dessous le code :

```
<!doctype html>

<html>

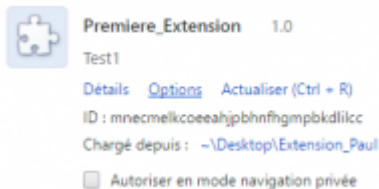
  <body>

    <h1>Options de l'extension</h1>
    <br>
    <h2>Saisissez le lien pronote de votre etablissement</h2>
    <label for="lien">Lien &nbsp;  </label><input type="text" name="lien"
id="link"/>
    <br>
    <br>
    <input id="btnenr" type="button" value="Enregistrer"/>
  </body>

  <script src="pageOptions.js"></script>
  <style>

</style>
</html>
```

Une fois cette page codée, vous devriez avoir la possibilité de cliquer sur Options lorsque vous vous rendez dans les extensions, sous Chrome :



Voilà, la page d'option est créée, il n'y a plus qu'à ajouter le fichier javascript.

L'interaction utilisateur - machine avec pageOptions.js

Afin de savoir ce que l'on va devoir faire lorsque l'utilisateur aura saisi l'URL, on va utiliser le javascript. Dans un premier temps, on va récupérer le lien, et le sauvegarder. Lorsqu'on a créé la page html, on a associé un ID au contenu de la textbox : link. On va l'utiliser pour récupérer ce contenu, avec la ligne suivante :

```
document.getElementById('link').value;
```

Bien-sûr, cette ligne n'a aucune utilité vu qu'on ne sauvegarde nulle part ce qu'on récupère. Pour le sauvegarder, on utilisera la fonction de stockage local, localStorage, qui permet d'enregistrer des données sur l'appareil de l'utilisateur, et qui se présente sous forme de tableau. Ce qui nous donne la ligne suivante :

```
localStorage['lien']=document.getElementById('link').value;
```

On voulait ensuite effectuer une redirection vers le lien saisi, et donc sauvegardé. Pour se faire, il suffit d'ajouter la ligne suivante, qui redirige directement le navigateur :

```
document.location.href= localStorage['lien'];
```

Malheureusement, nous n'avons pas terminé. Si on exécute le code comme ça, dès que la page d'option se lancera, il s'exécutera, et ne laissera pas à l'utilisateur le temps de saisir un lien. D'où l'utilité du bouton continuer. On va donc intégrer tout le code qu'on a fait dans une fonction, que l'on va nommer `modifLien` :

```
function modifLien()  
{  
    localStorage['lien']=document.getElementById('link').value;  
    document.location.href= localStorage['lien'];  
}
```

A la suite de ce code, on ajoute un écouteur qu'on associera au bouton continuer, et qui appellera la fonction `modifLien` dès que le bouton aura été cliqué :

```
document.getElementById("btnenr").addEventListener("click", modifLien);
```

Nous devrions donc avoir la page suivante qui s'affiche lorsqu'on clique sur Options :

Options de l'extension

Saisissez le lien pronote concernant votre établissement

Lien

Voilà le code final du fichier `pageOptions.js`, qui s'occupe de la gestion du bouton et du lien :

```
function modifLien()  
{  
    localStorage['lien']=document.getElementById('link').value;  
    document.location.href= localStorage['lien'];  
}  
  
document.getElementById("btnenr").addEventListener("click", modifLien);
```

On peut bien imaginer tout ce qu'il est possible de faire avec cette page, personnaliser une page web, en modifiant les polices d'écriture ou la couleur de fond, par exemple.

Retour projet 2017



- [PROJET 2017](#)

From:

<https://sioppes.lycees.nouvelle-aquitaine.pro/> - **APs et stages du BTS SIO du lycée
Suzanne Valadon**

Permanent link:

<https://sioppes.lycees.nouvelle-aquitaine.pro/doku.php/hackathlon/pronote/options>

Last update: **2017/06/05 20:19**

