

TME4 – Construction de l'architecture du site en React

Dans ce TME, nous nous concentrerons sur la page de connexion du site web. Elle inclue : un panneau de navigation où l'on pourra se connecter/déconnecter et un formulaire d'inscription.

Exercice 1 – Configuration sur les machines ppti

Q 1.1 Pour créer une application : `create-react-app nomdevotreapplication`. Vous aurez donc un dossier "nomdevotreapplication" qui va être créé avec des sources et fichiers de configuration. Pour commencer à travailler, se positionner dans le dossier de l'application et exécuter : `"npm start"`

Q 1.2 Si cela ne fonctionne pas, avant de commencer le TME, vous devez configurer votre machine (vérifier la procédure avec votre chargé(e) de TME) :

- Copier dans votre home à la racine de votre compte le fichier caché `.npmrc` hébergé dans `/Infos/react/config-file/.npmrc` : `cp /Infos/react/config-file/.npmrc .npmrc`
 - Créer dans votre homedir `~/` `.npm-global` : `mkdir .npm-global`
 - Créer ou ajouter au `.bashrc` la ligne suivante : `export PATH=~/npm-global/bin:$PATH`
-

Exercice 2 – Page principale

On s'intéresse en premier lieu à la structure du site web.

Q 2.1 Créer un composant `MainPage` qui initialise dans le constructeur l'état du composant qui permet de spécifier la page courante (connexion) et si l'utilisateur est connecté (au début "false").

Q 2.2 Créer la méthode `getConnected()` qui permet de modifier l'état du composant (page courante sera désormais le mur de tweets et l'utilisateur sera connecté). Afin de pouvoir utiliser cette méthode, il faut qu'elle puisse être appelée comme un callback ; et dans ce callback, `this` doit correspondre à l'objet courant... Pour cela, deux méthodes :

- Rajouter dans le constructeur la ligne de commande suivante : `this.getConnected = this.getConnected.bind(this);`. Cela permet de relier la méthode au composant `this` et de l'appeler facilement par la suite (voir <https://medium.com/shoutem/react-to-bind-or-not-to-bind-7bf58327e22a>).
- Utiliser la **syntaxe des champs de classe** :


```
getConnected = () => {
    // this.state correspond à l'objet
}
```

Q 2.3 Créer la méthode `setLogout()` qui permet de modifier l'état du composante (page courante : connexion et utilisateur non connecté). Relier la méthode au composant dans le constructeur.

Q 2.4 Créer la méthode `render()` qui pour l'instant contient seulement des balises `<div>`. Par la suite, on modifiera cette méthode pour afficher le panneau de navigation, le contenu de la page (connexion, mur de tweets ou page profil), ...

Q 2.5 Afficher ce composant en l'appelant dans le fichier `index.js` :
`ReactDOM.render(<MainPage/>, document.getElementById('root'));`

Exercice 3 – Panneau de navigation

Ce panneau de navigation contient, suivant l'état de connexion, le formulaire pour se connecter ou le bouton pour se déconnecter. **Q 3.1** Créer un composant `NavigationPannel` qui retourne, pour l'instant un paragraphe dans la méthode `render()`.

Q 3.2 Le composant `NavigationPannel` est appelé dans la méthode `render()` du composant `MainPage`. Lors de son appel, on fournira les attributs suivants : `login=this.getConnected` `logout=this.setLogout` `isConnected=this.state.isConnected`

Q 3.3 Modifier la méthode `render()` du composant `NavigationPanel` pour qu'elle retourne une balise `<nav>` qui contient soit le composant `Login` soit le composant `Logout`. Le composant `Login` contient comme attribut la méthode `getConnected()` définie dans le composant `MainPage`. De même pour le composant `Logout` qui contient l'attribut relié

à la méthode `setLogout()`.

Q 3.4 Créer le composant `Logout` qui affiche dans la méthode `render()` le bouton de déconnexion.

Q 3.5 Créer le composant `Login` qui affiche dans la méthode `render()` le formulaire de connexion (login, mot de passe et bouton).

Q 3.6 Vérifiez l’affichage de votre site web ! Changez l’état dans le composant `MainPage` (`this.state.isConnected` à `true` ou `false`) pour vérifier que les composants `Login` et `Logout` s’affichent bien.

Exercice 4 – Formulaire d’inscription

Q 4.1 Reprendre les étapes précédentes pour afficher le formulaire d’inscription en dessous du panneau de navigation :

- Création d’un composant `SignIn` ”vide”.
- Modification de la méthode `render()` dans le composant `MainPage` pour appeler le composant `SignIn`. Attention, cet affichage est conditionnel à la page courante spécifiée dans l’état.
- Modification de la méthode `render()` dans le composant `SignIn`.

Exercice 5 – La suite...

Q 5.1 Continuer le développement du client jusqu’à avoir le site complet (page mur de tweets, page profil, ...). Entre autres, il vous faudra faire un composant pour :

- Un message (si vous choisissez d’associer un message à un/des commentaires, il faudra implémenter les composants du commentaire, de la liste des commentaires, du formulaire de saisie de commentaire...)
- Une liste des messages
- Un formulaire de saisie de message
- Un ami
- Une liste d’amis
- Un utilisateur (ses informations, etc... pour l’afficher sur le bandeau de la page profil)
- Une page de profil
- Un mur de tweets
- ...