
CPE Lyon - 3IRC - Année 2018/19

Techniques et Langages d'Internet

TP 5 - AJAX, JSON, jQuery



Dans ce cinquième TP, vous manipulerez l'architecture AJAX, le format JSON et la bibliothèque jQuery.

Sujet du TP

Pour ce TP, nous aurons besoin d'un serveur HTTP. En effet, la plupart des navigateurs interdisent les requêtes AJAX en local pour des raisons de sécurité. Plusieurs solutions existent :

- si vous travaillez sur votre propre machine, vous pouvez installer WampServer (si vous êtes sous Windows) ou Lamp (si vous êtes sous Linux) ;
- si vous travaillez sur les postes de l'école, vous n'avez pas les autorisations pour installer ces logiciels ; vous pouvez alors soit passer par les machines virtuelles mises à votre disposition, soit utiliser un serveur web ne nécessitant pas d'installation.

Afin de ne pas perdre de temps avec l'installation d'un serveur web (que vous verrez plus en détail l'année prochaine), vous pourrez utiliser le serveur disponible en téléchargement sur **e-campus**. Ce serveur s'appelle **USBWebserver**, et a l'avantage de ne pas nécessiter d'installation, mais à l'inconvénient de ne fonctionner **que sous Windows**.

Son fonctionnement est très simple : il vous suffit de dézipper le fichier, puis d'exécuter le programme **usbwebserver**. Au bout de quelques secondes, le serveur HTTP (Apache) et le serveur de base de données (MySQL) devraient être démarrés :



Il vous suffit alors de saisir l'URL **localhost** dans un navigateur.

Vous trouverez également un autre zip contenant des dossiers à déposer sur votre serveur (dans le dossier **root** si vous utilisez UsbWebserver). Ces dossiers contiennent le *back-end* et des fichiers de ressources nécessaires aux exercices.

Vous penserez à user et abuser des outils de débogage proposés par les navigateurs, en particulier pour le débogage JavaScript et examiner les requêtes et réponses entre le navigateur et le serveur.

Exercice 1. Cours des cryptomonnaies

L'objectif de cet exercice est de réaliser une interface permettant d'afficher diverses informations (nom, symbole, cours...) sur les principales cryptomonnaies.

1. Utilisez une requête AJAX pour récupérer le fichier `cryptomonnaies/cryptomonnaies.json`.
2. Ecrivez une fonction JavaScript qui remplit une liste déroulante (objet HTML `select`) avec les noms des cryptomonnaies triés par ordre alphabétique (sans tenir compte de la casse).
3. Ecrivez une fonction qui met à jour la page web en affichant les principales informations (nom, symbole et cours) de la monnaie sélectionnée dans la liste.

Exercice 2. Messagerie instantanée

Le but de cet exercice est de réaliser une page web dotée d'une messagerie instantanée simple, limitée à deux utilisateurs. Chacun des utilisateurs parlera à l'autre en utilisant un onglet du navigateur, et en s'identifiant à l'aide de la variable `user` (vous aurez donc deux fenêtres, l'une avec l'URL `http://localhost/chat_ajax/?user=1` et l'autre avec l'URL `http://localhost/chat_ajax/?user=2`).

L'interface se composera au minimum d'une zone de texte affichant l'historique des messages envoyés, et d'une zone de saisie des messages, qui doit faire apparaître un texte explicatif (*placeholder*).

Les utilisateurs et les messages sont gérés dans une base de données MySQL livrée avec le serveur web. Vous devrez donc respecter les noms des objets indiqués dans la suite du sujet, faute de quoi vous risquez de ne pas voir vos messages !

1. Commencer par créer l'interface de votre « chat » dans le fichier `index.html` situé dans le dossier `chat_ajax` sur le serveur. Vous êtes bien entendu encouragés à utiliser des styles CSS si besoin (à placer dans le dossier `chat_ajax/css`).
2. Dans votre fichier JavaScript (à placer dans `chat_ajax/js`), créez un objet `chat` possédant les attributs et méthodes suivants :
 - un attribut `saisie` faisant référence à la zone de saisie du document HTML,
 - une méthode `posterMessage`,
 - une méthode `recupererMessages`.
3. Créez la méthode `posterMessage`. Cette méthode doit effectuer un appel AJAX au fichier `ajax/chat.php` et lui envoyer, avec la méthode POST, trois paramètres :
 - le nom de la méthode à appeler : `'poster'`, à renseigner dans une variable `method`
 - l'identifiant de l'utilisateur qui a saisi le message, dans une variable `user_id`
 - le contenu du message, dans une variable `message`
4. Créez la méthode `recupereMessages`. Comme à la question précédente, cette méthode doit effectuer un appel AJAX au fichier `ajax/chat.php`, en lui demandant cette fois d'exécuter la méthode `'recuperer'`. Cette méthode renvoie du text HTML préformaté prêt à être inséré dans la zone d'historique des messages.
5. Il vous reste à modifier l'attribut `saisie` afin de ne pas envoyer de message lorsque l'utilisateur appuie sur **Shift + Entrée** (ce qui lui permet d'envoyer des messages comportant des sauts de ligne). Vous pourrez également utiliser la méthode `preventDefault` pour enlever à la touche **Entrée** son comportement habituel (saut de ligne) et la réserver à l'envoi des messages.

Exercice 3. Météo

Créez-vous un compte sur le site www.openweathermap.org, et utilisez l'API de ce site pour récupérer et afficher sur votre site d'agence de voyage la météo des destinations que vous proposez.