

Semaine 8 : Structurer ses données en JSON et les récupérer à l'aide d'AJAX

Objectif :

L'échange de données est une problématique centrale en programmation. En effet tout l'intérêt d'un d'une application est de pouvoir exploiter des données. Pour l'instant, avec JavaScript, vous n'aviez aucun moyen de récupérer des données autres que celles rentrées par l'utilisateur. Dorénavant avec AJAX vous aurez la possibilité de demander des fichiers présents sur le même serveur ou dans certains cas des serveurs extérieurs, ce que l'on appelle des API. Cela va vous permettre de grandement dynamiser vos interfaces et aussi de faire un petit point sur la sécurité.

Bien évidemment pour échanger des données encore faut-il se les échanger dans le même format. C'est pourquoi nous reverrons le format JSON mais aussi la programmation orientée objet en JavaScript.

Compétences acquises :

- Installer un serveur local de développement composé de PHP7, Apache2 et MySQL
- Structurer ses données à l'aide du format JSON
- Créer une arborescence HTML sur la base de données au format JSON
- Programmer en orienté objet
- Utiliser le constructeur d'objets
- Effectuer des requêtes de type GET vers des fichiers stockés sur serveur
- Comprendre et utiliser des API
- Connaître et comprendre la faille XSS et CSRF

Ressources et exercices :

- Un parcours en français sur l'AJAX et également les API :

<https://openclassrooms.com/en/courses/3306901-creez-des-pages-web-interactives-avec-javascript/3626511-la-theorie-http-ajax-et-json>

- La documentation MDN :

https://developer.mozilla.org/fr/docs/Web/Guide/AJAX/Premiers_pas

Un récapitulatif synthétique en anglais

- https://www.w3schools.com/js/js_ajax_intro.asp

Projet fil rouge : Une application de gestion de comptes bancaires

Vous êtes développeur junior au sein du service informatique d'une grande enseigne bancaire. Le cœur de cible de cette banque était jusqu'à maintenant les épargnants âgés, qui utilisent peu internet. Elle souhaite maintenant diversifier sa clientèle, entrer de plein pied dans l'ère du numérique et proposer à ses usagers un service bancaire en ligne renouvelé afin d'attirer de nouveaux utilisateurs.

Cependant la banque est un métier de confiance et c'est la sécurité des utilisateurs qui doit primer avant tout.

A ce titre, vous devez créer une application qui permet à l'utilisateur de créer et gérer des comptes bancaires.

Spécifications fonctionnelles :

- Sur l'accueil du site, l'utilisateur voit par défaut tous ses comptes bancaires
- A son arrivée sur l'accueil un layer s'affiche par dessus l'ensemble de la page et lui rappelle les règles de sécurité essentielles sur un site internet. Les règles de sécurité sont stockées dans un fichier et récupérées par requête HTTP (AJAX).
- Sur une page de statistiques l'utilisateur accède à des informations bancaires comme les taux d'emprunt, le cours de la bourse, ect... Ces informations sont récupérées depuis un fichier via requête HTTP et présentées sous forme de tableau. Ces informations sont stockées dans un fichier au format JSON.
- Une page de blog, qui affichera des articles récupérés depuis l'API suivante : <https://oc-jswebsrv.herokuapp.com/api/articles>

/Optionnel si manque de temps/

- Sur une page dédiée, un formulaire lui permet de créer un nouveau compte bancaire avec minimum un type de compte (courant, livretA, PEL etc...) et une somme par défaut supérieure à 50 euros
- Pour chaque compte l'utilisateur peut cliquer sur un lien qui par la suite lui permettra de supprimer le compte
- Pour chaque compte, l'utilisateur peut, via un formulaire faire un dépôt d'argent
- Pour chaque compte, l'utilisateur peut, via un formulaire faire un retrait d'argent
- Sur une page dédiée, à l'aide d'un formulaire, l'utilisateur peut réaliser un virement d'un compte à un autre. Il peut donc sélectionner un compte A à débiter, indiquer un montant et sélectionner le compte B à créditer. Il ne peut sélectionner que ses propres comptes.

Spécifications techniques :

- HTML5
- CSS3
- Framework Bootstrap4
- Base Boilerplate
- JavaScript avec respect des normes ES6
- Vous avez produit des maquettes de type wireframe simples pour au moins une des pages
- Vous avez produit une arborescence fonctionnelle de l'application reprenant les cas d'utilisation possible de la page
- Vos wireframes sont accessibles dans un dossier DOC
- Votre interface est responsive sur tous les supports
- Vous respectez le principe DRY
- Votre code est commenté
- Votre code est hébergé sur GitHub
- Vous avez fait usage d'un logiciel de versionning
- Votre site est hébergé via une GH-page

- Le repository contient un readme, une description et des tags
- Vous utilisez un outil de gestion de projet de type KANBAN. Idéalement pour chaque tâche vous estimez le temps et la priorité et vous vous mettez dans la peau de l'utilisateur.

Le rendu se fera via github. Vous devrez déposer un fichier à votre nom sur TEAMS avant dimanche soir minuit.

Pour aller plus loin :

Vous pouvez ajouter des fonctionnalités à votre projet:

- Un formulaire de création et de connexion d'utilisateurs sur une page dédiée
- Rendre les formulaires fonctionnels en JavaScript. Par exemple si je remplis le formulaire de création de compte, les informations rentrées sont traitées en JavaScript et permettent de créer un nouveau compte en HTML qui est ajouté à la page. Pour cela vous devrez peut-être utiliser les sessions.
- Valider les formulaires en JavaScript