

PC3R - TME6: Micro-Projet Web (1/4)

Equipe Enseignante PC3R

28/02/2020

Objectif: Donner une description complète d'une application web à réaliser pendant la période des TMEs 6-7-8-9-10.

Rendu: Le rendu final pour ce TME doit être un unique document (.pdf, .md, .html): un *dossier* décrivant l'application.

1 Projet

Les séances 6-7-8-9-10 de TMEs sont dédiées à la réalisation d'une application Web en binôme.

Sujet et Fonctionnalités. Le sujet de l'application et les fonctionnalités proposées sont **libres**, tant que l'application respecte les contraintes données ci-dessous. Le nombre et la nature des fonctionnalités proposées doit rester raisonnable pour le temps imparti (micro-projet à rendre pour la dernière semaine de cours).

Contraintes.

- Le serveur doit être écrit en *Servlets* naturelles et doit s'exécuter dans Tomcat/Catalina **ou** doit être écrit en *Go* à l'aide du package `net/http`. Aucun *framework web* ne doit être utilisé.
- Pour le client, les bibliothèques *Javascript* usuelles (*jQuery*, *Bootstrap*, ...) sont autorisées. Les *framework client* de haut niveau (comme *Angular.js*) sont déconseillés (particulièrement aux débutants en Web) mais autorisés.
L'utilisation de *React* est encouragée.
- L'application doit proposer du contenu *généré par les utilisateurs*, un bon exemple est l'intégration d'une composante *sociale* (profils, commentaires, notes, messages, publications, ...).
- Le serveur de l'application doit contacter de manière régulière (en réaction à une requête, ou à des dates précises, par exemple "tous les soirs") une **API Web externe**, dynamique déjà existante (dynamique signifie que son contenu doit changer régulièrement, par oppositions aux API statiques) (exemple: *horaires SNCF*, *résultats sportifs*, *sortie de produits culturels*, ...).
- Les requêtes entre le client et le serveur sont majoritairement *asynchrones* (AJAX) et le serveur répond avec des données encodées en *JSON*.
- L'écriture d'une *API REST* pour la communication entre le client et le serveur est encouragée, mais pas obligatoire.

1.1 API Externe

Trouver une API externe qui répond à un besoin de l'application (généralement, l'aider à peupler, et mettre à jour, sa base de données). Utiliser les sites qui listent des APIs (*Programmable Web*, gouvernement <https://api.gouv.fr/>, *SNCF* <https://ressources.data.sncf.com>) ou vérifier la présence d'une API Web attachée à un site connu (sites de ventes en ligne, de transports, de météo, de nouvelles, *Google Maps*, ...).

Vérifier les conditions d'utilisation de l'API: besoin ou non de demander une clef, nombre limite de requêtes par jour, fréquence de mise-à-jour des données, ...

Simuler à la main des appels à l'API pour étudier le format des données récupérées.

1.2 Sujet

Essayer de trouver un sujet original d'application utilisant une telle API. Dans le pire des cas, se rabattre vers les classiques "un réseau social autour de ...", "un site de critiques et d'évaluations de ...", "une agence de rencontre pour ...", *etc*

2 Dossier

Le dossier à rendre, pour la veille du TME7, doit contenir:

- un paragraphe décrivant le sujet de l'application et ses principales fonctionnalités,
- un paragraphe décrivant l'API Web choisie, le contenu qu'elle propose et le contenu que l'application du projet utilisera (et comment elle l'utilisera),
- un paragraphe décrivant les fonctionnalités de l'application,
- quelques paragraphes décrivant, de *manière textuelle*, des cas d'utilisations classiques de l'application, présentant ses diverses fonctionnalités.
- un liste des données qui seront stockées dans la base de données du serveur, leur composition, éventuellement des schémas de tables.
- un paragraphe expliquant comment les données sont mises à jour et quand l'API externe est appelée,
- une description du serveur: choix de l'approche (Service ou Ressources), composition (liste des services ou des ressources) et fonctionnalités associées à chacun des composants.
- une description du client: plan du site (probablement une application monopage), contenu des écrans ; bien identifier à quels endroits apparaissent les appels aux différents composants du serveur,
- une description des requêtes (*query string* ou requête HTTP) et réponses (contenu JSON de la réponse HTTP) entre le client et le serveur,
- un schéma global du systèmes, présentant un résumé des informations du dossier.