



Mini projet LW1 – sCiMS – Consignes finales

M1 Informatique

(Client : florent.nicart@univ-rouen.fr)

6 décembre 2016

1 Rappel des fonctionnalités attendues

Le projet consiste en l'écriture d'un mini CMS pour la mise en ligne d'articles à caractère scientifique.

Le site sera divisé en catégories qui peuvent être imbriquées les unes dans les autres pour organiser le contenu. Chaque catégorie disposera de sa propre URL qui aura pour effet d'afficher la liste de ses sous-catégories ainsi que les 10 (par exemple) derniers articles ajoutés dans cette catégorie. La racine du site correspond à l'affichage de la catégorie racine.

Les articles seront structurés comme indiqué à la fiche de TP N°4.

On distinguera trois types de rôles pour les utilisateurs :

- Anonyme : peut seulement visualiser le contenu des catégories et des articles,
- Rédacteur : un utilisateur authentifié qui peut créer des articles, modifier ou supprimer les articles dont il est propriétaire.
- L'administrateur : peut ajouter/supprimer/modifier des utilisateurs et des catégories à travers une interface d'administration qui lui sera réservée. (De manière général, il n'est pas souhaitable qu'un administrateur cumule d'autres rôles, ici celui de rédacteur).

2 Coopération LW1/Compilation : le traitement des formules \LaTeX

Rappel : les articles peuvent contenir des formules mathématiques qui seront saisies par l'utilisateur en langage \LaTeX , plus pratique que \MathML mais dont l'inclusion et le rendu sera assuré par \MathML .

Pour alléger la charge de travail, il a été convenu que cette tâche de conversion serait l'objet de votre mini projet de compilation. Vous avez cependant la possibilité d'utiliser une autre approche si vous le souhaitez, mais cette charge de travail supplémentaire ne doit pas se substituer à la bonne réalisation des fonctionnalités demandées.

Pour rappel, voici l'approche proposée pour la modification :

- lorsqu'il est en cours d'édition d'un article, un rédacteur peut éditer une formule en cliquant sur l'élément contenant le rendu de la formule ;
- un champs de saisie contenant le code \LaTeX original de la formule lui est proposé ;
- il peut y altérer le code \LaTeX et a accès à deux fonctions : **annuler** et **enregistrer** ;
- lorsqu'il enregistre son code \LaTeX , celui-ci est envoyé à un service Web (en \AJaX) qui invoque le programme de votre projet de compilation, récupère le résultat et renvoie le flux \MathML produit ou un code d'erreur en cas de syntaxe incorrecte.
- en cas d'erreur, un message est affiché et la saisie se poursuit,
- sinon, le flux \MathML renvoyé vient remplacer celui du rendu déjà en place et le nouveau code \LaTeX est caché à l'intérieur pour les modifications futures.

Puisque votre projet de compilation ne sera probablement pas disponible à la livraison du projet Web, vous placerez un bouchon, c'est à dire un programme compilé qui renvoie une réponse constante ou bien triviale pour des tests. (Pensez à fournir le code source de ce programme).

3 Contraintes de réalisation

- Vous êtes entièrement libre dans le choix des langages (Php, Java, Haskell). Cependant, vous ferez en sorte que votre livrable soit facilement installable. En particulier, pour les choix exotiques, vous fournirez un manuel installation pour la résolution des dépendances particulières de votre projet (exemple : interpréteur, bibliothèques, voire configuration particulière d'Apache, etc.).
- Les frameworks sont autorisés à tous les niveaux – MAIS – pour chaque framework utilisé, vous vous emploierez dans votre rapport à démontrer que vous vous êtes approprié la philosophie du framework. Vous démontrerez notamment que :
 - vous faites un emploi conforme du framework,
 - vous respectez les principes vus en cours,
 - vous maîtrisez le langage sous-jacent (Php, JavaScript, CSS, etc.).
- Idéalement, l'installation doit se résumer à déposer une archive sur un serveur Web puis à lancer une interface de configuration (pour configurer l'accès à la base de donnée pré-existante par exemple). Si vous n'y parvenez pas, la modification d'un **unique** fichier de configuration (bien documenté dans votre manuel) sera acceptable.
- Il est souhaitable de livrer votre application avec des données de test afin de pouvoir la tester avec du contenu. (Notez que ceci est valable pour vos projets annuels également !)
- **Avertissement** : les projets qui ne fonctionnent pas correctement une fois installés ne seront pas corrigés pour être évalués. Nous ne pouvons pas nous permettre de passer deux heures sur chaque projet juste pour essayer de les faire fonctionner. Il vous est donc fortement recommandé, avant de les livrer, de tester leur déploiement sur une machine vierge conformément à votre manuel d'installation afin d'éviter, entre autres :
 - les bibliothèques manquantes,
 - des réglages spécifiques du serveur que vous auriez oublié de documenter,
 - les étapes de configuration manquante dans le manuel,
 - des erreurs de capitalisation dans les noms de fichiers/bibliothèques/tables de bases de données, etc (pour ceux PERSistent à traVAILLER Sous windows !;-)).Pour vous aider, une machine virtuelle (*VirtualBox*) d'un mini serveur vous est fournie à cette adresse : <https://ent.normandie-univ.fr/filex/get?k=0Y7154pbLVmPdNBAEj0>. Compte tenu de la taille, il vous est conseillé de la télécharger depuis la Fac.
- Pour finir, votre rapport doit servir à valoriser votre apprentissage au cours de la réalisation de ce projet. Les copier-collers de *Wikipédia* ou autres sites Web n'ont aucun intérêt (et se repèrent facilement).

4 Notice d'utilisation rapide de la machine virtuelle

Une fois *VirtualBox* installé, la machine virtuelle peut être utilisée en exécutant les étapes suivantes :

- Ouvrez le menu **Fichier**→**Paramètres**, puis allez sous l'onglet **Réseau**→**Réseau hôte uniquement**.
 - Ajoutez un réseau **vboxnet0** puis éditez le pour qu'il corresponde aux réglages suivants :
 - Carte/adresse IPv4 : 192.168.76.1 (masque 255.255.255.0).
 - Serveur DHCP : adresse 192.168.76.100/255.255.255.0, limites 192.168.76.100 à 192.168.76.254.
 - Fermez ce menu.
 - Importez ensuite l'image depuis **Fichier**→**Importer un appareil virtuel**, choisir le fichier **.ova**.
- Une fois importée, la machine peut être démarrée. vous disposez alors :
- D'une machine accessible sur l'adresse ip 192.168.76.76.
 - D'un compte utilisateur **urouen:madrillet**,
 - accessible en *ssh* depuis **urouen@192.168.76.76**,
 - et en *sftp* pour déposer des fichiers depuis un gestionnaire de fichiers avec l'URL **sftp://urouen@192.168.76.76**.
 - D'un serveur web avec un dossier accessible en écriture à votre utilisateur sous **/var/www/html/projet** servi par *Apache* sur **http://192.168.76.76/projet/**.
 - D'une base de données accessible par vos applications web avec les paramètres suivants :
 - nom de la base de données (schéma) : **projet**,
 - adresse du serveur : **localhost**,
 - nom de l'utilisateur : **projet**,
 - mot de passe de l'utilisateur : **tejorp**.
 - D'un serveur *tomcat* accessible sur **http://192.168.76.76:8080** et
 - d'une interface d'administration *tomcat* accessible sur **http://192.168.76.76:8080/manager** avec l'utilisateur **tomcat:tomcat**.

- Le compte système `root` est accessible avec le mot de passe `rotomagus`.
- Une interface *phpmyadmin* est également disponible sur `http://192.168.76.76/phpmyadmin`.

Inutile de préciser qu'étant donné la robustesse de la configuration de cette machine, elle est destinée à faire des tests et en aucun cas à être mise en ligne!