

Projet programmation web

2014-2015

rgiraud@labri.fr

Introduction

Ce projet de programmation web IT103 s'effectue par binômes. Il a pour but de vous faire développer une application web utilisant l'architecture dynamique côté serveur.

Vous serez notés sur toutes les composantes de l'application :

- Qualité du code HTML/CSS (/6)
- Qualité de l'apparence visuelle, simplicité d'utilisation (/4)
- Qualité du code PHP/SQL, qualité de la structure de la base de données (/6)
- Facilité de déploiement (/4)

Voici la liste des technologies qui doivent être utilisées dans le projet :

- HTML5
- CSS 3
- MySQL
- PHP 5, fonctions uniquement, pas de mécanismes objets

Déroulement

Les groupes sont constitués une fois pour toutes au début de la première séance. Le reste de la séance sera consacré à la clarification du sujet et au début de l'implémentation. Une deuxième séance sera ensuite effectuée pour vérifier que tout le monde a bien compris le sujet et pour répondre aux questions techniques qui auront été soulevées par une première confrontation au sujet.

Rendu

Concernant le rendu, l'archive du projet contenant tous les fichiers est à uploader sur moodle (cours IT103). Le code source du projet est demandé. Celui-ci sera situé dans un répertoire *src*. Pas de rapport pour ce projet, seulement un fichier au format texte brut, nommé *architecture.txt*, décrivant l'intégralité des fichiers contenus dans votre application, selon la forme suivante :

```
src/fichier1.php : description du fichier 1.  
src/fichier2.js : description du fichier 2.  
...
```

Les descriptions doivent être courtes et indiquer clairement quel est le rôle du fichier. Enfin, vous devez fournir un document en texte brut nommé *installation.txt* qui explique la procédure d'installation, si nécessaire, de votre application web (initialisation des bases de données, etc...).

Le tout est à compresser dans une archive nommée *it103-2015-gN.zip* avec *N* étant le numéro de votre groupe attribué sur moodle. Veillez à bien indenter vos fichiers de code PHP, CSS, HTML, etc. Les groupes ne respectant pas ces règles ou celles de la section déploiement seront pénalisés lors de la notation. Le projet est à uploader sur moodle au plus tard le 20/05/2015.

Enfin faites particulièrement attention à la copie, j'utilise un logiciel qui permet de détecter les copier-coller automatiquement, tout projet suspect sera étudié avec la plus grande attention...

Sujet

Vous devez développer une application/site de brocante en ligne. Le principe de cette application est le suivant : il s'agit de pouvoir mettre en ligne les photos en vrac de son stand ainsi que quelques descriptions. Contrairement aux sites type « leboncoin », ici les photos sont des photos globales des stands, comme dans les vraies brocantes. Plus simple pour les utilisateurs (vendeurs) que de décrire chaque objet et de fixer un prix. Aux autres utilisateurs (acheteurs) ensuite de fouiller les photos des brocantes à proximité, à la recherche de l'objet rare, et de contacter le vendeur si un objet les intéresse.

Pour créer leurs brocantes et uploader des photos, les utilisateurs doivent d'abord s'inscrire ; ensuite ils peuvent se connecter sur l'application ; après connexion, ils peuvent éditer leur profil, ajouter des photos, descriptions, etc. En revanche, il n'est pas nécessaire de s'inscrire pour visualiser les brocantes en ligne.

Fonctionnalités

Écran d'accueil/connexion :

L'écran d'accueil est un simple écran présentant le site. Son fonctionnement, ses règles, les liens vers les différents menus, etc, soyez inventifs. Celui-ci peut suggérer à l'utilisateur de se connecter. Pour cela, ce dernier doit fournir un mail ainsi qu'un mot de passe. L'écran de connexion doit fournir un lien vers l'écran d'inscription. Si l'utilisateur arrive à se connecter, il doit être redirigé vers son l'écran de son profil. Sinon, un message d'erreur doit expliquer pourquoi la tentative de connexion a échoué.

Écran d'inscription :

L'écran d'inscription permet à un utilisateur de s'inscrire. Pour cela il doit fournir un mail, un mot de passe qui doit être saisi deux fois pour éviter les erreurs. Il doit aussi fournir un nom, un prénom, un pseudo, un département, et une ville. Une fois l'utilisateur inscrit, il doit être redirigé vers l'écran de connexion. Attention, il ne peut y avoir qu'un seul utilisateur ayant une adresse mail donnée. Les vérifications sur les champs du formulaire doivent être effectuées côté client et côté serveur.

L'utilisation d'un script Javascript est possible et recommandée (mais facultative car non vu en cours) pour vérifier avant l'envoi, l'état du formulaire. Les vérifications à effectuer peuvent être les suivantes : le mot de passe fait au moins 6 caractères, les noms font plus de deux caractères, le département est valide, le mail est dans un format correct (login@domaine.extension), et toutes les données demandées ont été saisies. Si les données saisies par l'utilisateur ne sont pas correctes, elles peuvent être affichées sur fond rouge. Dans ce cas des messages, également en rouge, peuvent expliquer pourquoi les données ne sont pas correctes.

Écran de profil :

Cet écran permet de visualiser le profil d'un utilisateur. Description, lieu de résidence, liens vers ses différentes brocantes, ou encore photo (optionnel). Bien sûr la modification des données ne doit être rendue possible que pour l'utilisateur lui-même.

Écran des brocantes :

Un utilisateur peut créer jusqu'à 3 brocantes, thématiques ou non. Dans chacune d'elle il peut uploader jusqu'à 2 photos de 2Mo max. Lorsque l'on clique sur une brocante, on voit donc le détail des photos, et une description associée. Un lien vers le profil du vendeur est bien sûr disponible, ainsi qu'un lien « contacter le vendeur ».

Pour l'upload de photos, regarder du côté de :

<http://openclassrooms.com/courses/upload-de-fichiers-par-formulaire>



Figure 1 : Exemple synthétique d'agencement de l'écran Brocante.

Écran de mail :

Comme pour « leboncoin », vous pouvez envoyer un mail au vendeur concernant une brocante (demande de prix d'un objet, photos supplémentaires, etc.). L'adresse mail du vendeur ne doit pas être visible et le nom de la brocante doit être passée en argument et pré-remplie dans l'objet du mail.

Écran de recherche :

Ici vous pouvez avoir un aperçu des différentes brocantes en ligne. Une présentation synthétique doit donc être faite, type première photo, début de la description, etc. Parmi les options de recherche, l'utilisateur doit pouvoir sélectionner une région ou un département. Cette information est sélectionnée depuis le champ département renseigné lors de l'inscription de l'utilisateur qui a mis en ligne sa brocante.

Exemple : J'habite en Gironde, je crée 1 brocante, que je nomme « chez TOTO ». Un autre utilisateur dans le département a créé une brocante « A la bonne affaire » dans laquelle il stocke un tas d'antiquités en tout genre. Si je souhaite créer une seconde brocante, je vais sur mon profil et je crée une nouvelle brocante. Celle-ci apparaîtra alors dans les derniers résultats de recherche ainsi qu'en Gironde et en Aquitaine.



Figure 2 : Exemple synthétique de configuration de l'écran de Recherche.

Fonctionnalités bonus :

A vous d'être imaginatifs, inspirez-vous des autres sites. Chaque fonctionnalité en plus sera prise en compte et appréciée selon sa difficulté.

Liste non exhaustive :

- Tri des brocantes selon la date d'ajout, le nombre de photos. (++)
- Tri selon plusieurs paramètres. (+++)
- Enregistrer des brocantes en favoris. (++)
- Choix pour l'utilisateur de la photo à mettre en avant. (+++)
- Liens utiles (tarifs de la poste, etc.). (+)
- Utilisation de JavaScript pour la vérification des données, champs, etc. (++++)

Indications :

Bien évidemment, le contenu variable du site (utilisateurs, descriptions) devra être stocké dans une base de données de type MySQL. Réfléchissez bien en amont à la manière de structurer votre base de données. Celle-ci est en effet plus complexe que celle abordée dans les TP. A l'ENSEIRB, vous avez la possibilité de développer du code PHP et MySQL depuis votre page *vvvpedago*. Votre base de données peut être gérée depuis *phpmyadmin*.

A noter qu'une seule base au nom de votre login est disponible et que les droits d'accès ne sont pas paramétrables. Pour vous y connecter depuis un script php (utilisation de *mysqli*),

```
$mysqli = mysqli_connect ('localhost', 'login', 'password', 'login') ;
if (mysqli_connect_errno($mysqli)) {
    echo "Echec lors de la connexion à MySQL : " . mysqli_connect_error();
}
```

avec *password*, le mot de passe que vous avez reçu par mail le jour de la création de votre page web. Pour récupérer la base de données, si vous travaillez via *phpmyadmin* sur les machines de l'école, vous devez exporter votre base. Pour cela, placez-vous sur *localhost* et suivez la procédure d'exportation pour récupérer votre *localhost.sql* qui sera à renommer avant rendu (voir section déploiement).

Bien entendu, le mot de passe des utilisateurs ne doit jamais être stocké en clair dans la base de données. Vous devez donc trouver un mécanisme qui évite cela (fonctions de hachage).

Toute la partie serveur (réception des données provenant des formulaires, écritures et lectures dans la base) doit se faire en PHP. Pour être à même de gérer le mécanisme de connexion (qui nécessite de pouvoir savoir à la demande d'une page si on est déjà connecté ou non pour afficher l'écran de connexion ou l'écran d'accueil) vous devez utiliser les sessions PHP.

Les pages web doivent être en HTML et ne pas renvoyer d'erreurs quand elles sont validées par l'outil du W3C. CSS2 doit être utilisé le plus possible et doit aussi ne pas renvoyer d'erreur après validation pour l'outil du W3C.

Enfin, les formulaires peuvent être validés en utilisant Javascript. Vous avez le droit de vous appuyer sur la bibliothèque JQuery (non abordée en cours, donc auto-formation) pour simplifier l'écriture du code Javascript. Pour vérifier la correction des données rentrées par l'utilisateur (est-ce que c'est bien une adresse mail, etc.), je vous conseille d'utiliser les expressions régulières. Elles sont utilisables autant en Javascript que en PHP. Les vérifications sur les champs du formulaire doivent se faire évidemment côté client et côté serveur afin d'éviter des problèmes de sécurité. En outre vous pouvez utiliser Javascript pour la vérification des données quand c'est possible. L'utilisation de JavaScript pour la vérification est ici optionnelle, mais en pratique pour vos futurs applications, TOUJOURS effectuer la vérification les données ! (Autant vous former lors de ce projet).

Déploiement :

L'archive du projet contenant tous les fichiers est à uploader sur moodle (cours IT103). Avant de rendre votre projet, que vous ayez développé sur vos machines ou celles de l'ENSEIRB, vous devez modifier les droits d'accès à votre base de données MySQL (sinon sanction !).

Votre fichier est donc à renommer : *gN.sql* et la base accédée dans ce fichier doit se nommer *gN*.

Pensez également à modifier dans tous vos fichiers, vos procédures de connexions à la base ! Effectivement, lorsque je vais récupérer vos bases, que j'importe sur ma machine, les connexions doivent se faire automatiquement via,

```
mysqli_connect('localhost', 'root', '', 'gN')
```

Pensez à fournir une base de test suffisante (utilisateurs de différents départements, plusieurs brocantes chacun, etc.) afin que les fonctionnalités puissent être directement testées.

A la décompression de l'archive et après import de votre base, un double-clic sur votre page d'accueil (.php) doit permettre de tout tester. Cet aspect sera pris en compte dans le barème de notation.

Conseils :

La programmation Web demandée est assez vaste mais relativement facile, il s'agit donc beaucoup de s'autoformer. Le travail demandé n'est pas difficile, il suffit d'aller chercher l'information, de trouver des exemples. La question que vous vous posez a forcément déjà été posée en ligne, alors utilisez Google !

Un trou sur l'écriture de tel ou tel fonction : <http://www.labri.fr/perso/falleri/perso/ens/it103/> pour une réponse rapide.

Pour ceux qui veulent développer sur leur machine, je recommande l'utilisation de XAMPP qui permet d'installer un serveur web avec PHP en local avec tout ce qu'il faut !

<https://www.apachefriends.org/fr/index.html>