**PROJET IO2 2010**

**OBJECTIF**

L'objectif du projet est de créer une petite plate-forme permettant la publication de messages sur une page web. Votre site devra comporter les fonctionnalités suivantes :

* Chaque message est composé d'un titre, d'un nom d'auteur et du texte du message.
* Votre page principale affichera les messages déposés par les utilisateurs. Pour faciliter la navigation, cette page devra aussi présenter un menu rappelant les titres de tous les messages affichés sur la page, et permettant d'y accéder directement.
* Les messages seront stockés dans une base de données gérée via MySQL.
* L'ajout d'un message par un utilisateur se fera à l'aide d'un formulaire envoyé à PHP.
* La navigation dans votre site doit être aisée : servez vous de CSS pour donner à votre site une mise en page de qualité professionnelle (au moins un header, une colonne pour le menu et une pour le contenu). Rendez facilement identifiables les différents éléments de la page et chaque message. Il vous est interdit de vous servir des tableaux HTML pour faire de la mise en page.
* Au moins une fonctionnalité supplémentaire (voir dernière section). Choisissez cette fonctionnalité au début du développement car elle pourrait influencer fortement la structure de votre site.

**EVALUATION**

La qualité de votre projet sera évaluée en fonction des critères suivants :

* Correction et respect des standards : nous attendons du code valide, respectant le standard XHTML Strict : pensez à valider vos pages! Soyez aussi attentifs à la compatibilité de votre encodage : vous êtes libres d'utiliser l'encodage que vous préférez (on vous conseille, bien sûr, UTF-8), mais celui-ci doit être cohérent sur l'intégralité du site et il ne doit pas y avoir d'erreurs d'affichage.
* Compatibilité : pensez à tester votre site sur plusieurs systèmes (Windows, BSD, Linux, Mac, ...) et navigateurs (Firefox, IE, Konqueror, Safari,...).
* Clarté, lisibilité, concision : l'évaluation de votre travail sera basée sur votre code. Celui-ci doit donc être facilement lisible, aéré, commenté, indenté et concis. N'oubliez pas que du code sur internet est fait pour être partagé.
* Utilisation des connaissances : votre projet reflétera votre compréhension et votre recul sur les TPs que nous avons faits. Nous serons sensibles aux efforts que vous ferez pour faire apparaître les points intéressants que nous avons étudiés en HTML, CSS, PHP, MySQL (et pourquoi pas Javascript).
* Investissement, originalité : vous devez vous approprier le sujet et nous proposer un résultat qui mette en valeur votre travail. Plus vous vous investirez personnellement, plus vous obtiendrez ce résultat.

**ORGANISATION**

Ce projet sera effectué par binôme. Nous n'imposons pas les groupes de binômes, mais nous vous conseillons de choisir un(e) étudiant(e) d'un niveau équivalent au vôtre dans votre groupe de TP : pensez bien que nous saurons de toute façon évaluer le travail et l'investissement de chacun dans ce projet.

Les soutenances de projets auront lieu entre le 3 et le 14 mai. La soutenance se partagera entre une courte présentation de votre projet et des choix que vous aurez faits, et une discussion avec les moniteurs chargés de vous évaluer. Notez que la période de développement de votre projet est courte et nous vous conseillons de commencer à y réfléchir dès à présent.

Nous vous encourageons à poser toutes les questions qui peuvent faire avancer votre projet. Nous ne pénaliserons en aucun cas les étudiants ayant fait la démarche de se renseigner sur les moyens de dépasser des problèmes rencontrés dans la mise en oeuvre de leur projet. Au passage, nous vous rappelons les trois meilleurs moyens de répondre à vos questions : w3schools, les moniteurs, le forum sur le didel du cours d'IO2. Soyez prudents dans l'utilisation d'autres moyens d'information.

**ALLER PLUS LOIN**

Il vous est demandé d'ajouter au moins une fonctionnalité supplémentaire ; vous êtes bien sûr autorisés, et même invités, à en ajouter plusieurs.

Voici une liste de fonctionnalités possibles :

Gestion des utilisateurs

vous pourrez développer une base de données des utilisateurs (en utilisant des mots de passe) permettant de contrôler que le nom de l'auteur d'un message n'est pas usurpé. Vous pourrez faire en sorte qu'un utilisateur puisse corriger ses messages. **Prérequis** : Sessions PHP, MySQL.

Messages personnels

les messages auront aussi un destinataire (ou plusieurs) ; seul(s) le(s) destinataire(s) pourront lire les messages qui leur sont adressés. **Prérequis** : Gestion des utilisateurs.

Affichage par auteur ou par thème

permettez au lecteur de choisir un auteur et de ne voir s'afficher que les messages de cet auteur. Vous pouvez aussi demander aux utilisateurs de classer leurs messages par thème et permettre au lecteur de ne consulter que les messages se rapportant à un thème donné. **Prérequis** : MySQL.

Niveaux de réponse

pour plus de lisibilité, on voudrait pouvoir répondre spécifiquement à un message. Les réponses à un message pourraient ensuite s'afficher immédiatement après celui-ci, avec un décalage pour marquer la réponse. Vous pourrez faire en sorte qu'il y ait plusieurs niveaux de réponse. **Prérequis** : un peu d'algorithmique.

Animations jQuery

servez-vous de la bibliothèque jQuery pour ajouter de la dynamique à votre page. Par exemple vous pouvez faire en sorte que les messages puissent être ouverts ou fermés en cliquant sur le titre, vous pouvez permettre de réordonner les messages, etc. **Prérequis** : Javascript.

Styles multiples

fournissez plusieurs feuilles de style qui présentent des designs fondamentalement différents (se limiter à changer les couleurs serait considéré quelque peu insatisfaisant!). Ensuite, permettez à l'utilisateur de choisir quelle feuille appliquer : vous avez le choix entre PHP ou Javascript pour ce dernier point. **Prérequis** : CSS, et l'un entre Javascript et PHP.

Sécurité

assurez-vous de la robustesse de votre page, en essayant de prévoir des remparts à un utilisateur mal intentionné. En particulier faites attention à l'injection de données (XSS) dans les messages et les bases de données. Les **références** suivantes peuvent vous donner une idée du problème:

* [http://fr.wikipedia.org/wiki/Cross-site\_scripting](http://fr.wikipedia.org/wiki/Cross-site_scripting%20)
* [http://fr.wikipedia.org/wiki/Injection\_SQL](http://fr.wikipedia.org/wiki/Injection_SQL%20)
* <http://www.apprendre-php.com/tutoriels/tutoriel-39-introduction-aux-cross-site-request-forgeries-ou-sea-surf.html>
* [http://fr.wikipedia.org/wiki/Cross-site\_cooking](http://fr.wikipedia.org/wiki/Cross-site_cooking%20)
* <http://www.ghostsinthestack.org/article-8-les-bases-des-injections-sql.html>
* [http://www.cert.org/advisories/CA-2000-02.html](http://www.cert.org/advisories/CA-2000-02.html%20)
* [http://www.cert.org/tech\_tips/malicious\_code\_mitigation.html](http://www.cert.org/tech_tips/malicious_code_mitigation.html%20)
* [http://www.kb.cert.org/vuls/id/867593](http://www.kb.cert.org/vuls/id/867593%20)
* <http://www.securiteam.com/securityreviews/5EP0L2KHFG.html>

**Prérequis** : aucun en particulier, mais une bonne compréhension globale du cours et du protocole HTTP.

Voici quelques autres idées d'un niveau plus avancé qui vous permettront de découvrir des techniques qui n'ont pas été abordées en cours :

Fils RSS

ajoutez des fils RSS à votre page qui seront mis à jour à chaque publication d'un nouveau message. Créez au moins un fil pour tout le site et un fil par auteur.**Références** : <http://www.w3schools.com/rss/default.asp>. **Difficulté : \*\***

Éditeur WYSIWYG

transformez l'interface pour rentrer les messages en un éditeur WYSIWYG permettant d'insérer des éléments basiques de mise en forme comme du gras, de l'italique, etc. À cette fin vous pouvez écrire votre propre code Javascript ou vous servir d'une bibliothèque déjà écrite comme tinyMCE. Notez bien que les deux exercices sont totalement différents et que nous ne nous attendrons pas à ce que les résultats soient comparables. **Références** :<http://tinymce.moxiecode.com/>. **Difficulté : \*\*\* ou \***

Intéraction de type AJAX

à l'aide de Javascript et de l'objet XMLHttpRequest, vous pouvez faire en sorte que l'utilisateur soit prévenu en temps réel des nouveaux messages postés par les autres utilisateurs. En poussant l'idée jusqu'au bout, vous pouvez transformer votre page en une webapp de messagerie instantanée dans le même esprit que Meebo. Par souci de compatibilité, nous vous conseillons fortement d'utiliser des bibliothèques Javascript qui implantent une surcouche AJAX, plutôt que d'utiliser XMLHttpRequest directement; vous pouvez vous servir, par exemple, de jQuery ou de tw-sack. Les **références** suivantes pourront vous aider :

* [http://www.scriptol.fr/ajax/ajax-garrett.php](http://www.scriptol.fr/ajax/ajax-garrett.php%20)
* <http://dcabasson.developpez.com/articles/javascript/ajax/ajax-autocompletion-pas-a-pas/>
* [http://api.jquery.com/category/ajax/](http://api.jquery.com/category/ajax/%20)
* <http://www.twilightuniverse.com/?page_id=243>

**Difficulté: \*\*\***

Vous pouvez, enfin, nous proposer vos propres idées. Parlez-en impérativement à votre moniteur avant de vous lancer : il saura vous dire si votre idée est trop légère ou trop difficile par rapport à ce que nous demandons.

**MODALITÉS DE REMISE**

Vous devez remettre par mail à votre chargé de TP les sources de votre projet ainsi qu'une court compte rendu sur ce que vous avez réalisé. Le compte rendu doit rapporter au moins

* Une description de votre projet : quel est son but, quelles sont ses caractéristiques, comment l'utilisateur se sert de l'interface.
* L'architecture de votre site : quel sont ses composants, quels rôles ils accomplissent, comment ils sont distribués dans les fichiers source, à quelles bibliothèques externes vous avez fait recours, quelle est la structure de la base de données.
* Comment vous vous êtes partagés le travail, quelles difficultés vous avez rencontrées.

Le compte rendu peut être rédigé dans un fichier texte ou html ou word processor (Word, Open Office) ou pdf ou ps. La méthode préférée est le fichier html, possiblement faisant partie intégrante de votre site (par exemple joignable par un lien "À propos" en bas de page).

Vous devez rendre les fichiers source et le compte rendu dans un unique fichier compressé (zip, rar, tar, gzip, bzip). Il doit contenir :

* Tous les fichiers php, html, css, les images, etc.
* Le compte rendu,
* Si possible, un extrait de la base de données au format SQL dans un ou plusieurs fichiers .sql. Pour ce faire, vous avez deux possibilités : en ligne de commande avec la commande

mysqldump -u [login] -h pams.script.univ-paris-diderot.fr -p --databases [login] > base.sql

ou via phpMyAdmin dans l'onglet "Export" (allez voir [ce lien](http://www.enseignement.polytechnique.fr/informatique/INF441/INF441a/Session1/TD/TD3/TD3.php), section 3.4, pour un exemple de comment on exporte une table).