

24h des IUT Informatique

6-7 avril 2012 – Montreuil

Epreuve de programmation Web

MiniWiki

« Un **wiki** est un **site web** dont les **pages** sont modifiables par les visiteurs afin de permettre l'écriture et l'illustration **collaboratives** des documents numériques qu'il contient. », [<http://fr.wikipedia.org/wiki/Wiki>]

Description générale

L'objectif de l'épreuve est de construire un moteur de wiki. Ce moteur doit permettre aux visiteurs de créer des pages wiki et de les parcourir. Comme tous les moteurs wiki, MiniWiki permet aux visiteurs :

- de parcourir le contenu en suivant des liens,
- d'éditer chaque page pour en modifier le contenu, en utilisant une syntaxe simplifiée,
- de rechercher des pages,
- d'accéder à l'historique des modifications apportées à chaque page.

MiniWiki utilise la syntaxe MediaWiki pour l'édition des contenus. Le langage de développement est libre : java, ruby, python, php ... Pour des raisons de commodité, les exemples dans la suite sont donnés en php.

Description détaillée : fonctionnalités minimales attendues

Une page MiniWiki est identifiée par son nom. Ce nom est donc unique. Le nom d'une page est une chaîne de caractères pouvant contenir les caractères alphabétiques et numériques classiques, ainsi les 2 caractères `_` -

Deux fichiers contenant le texte source de 2 pages wiki vous sont fournis : `Main.txt`, `PageListes.txt`. Ils vous permettent de tester votre wiki.

Le moteur miniWiki doit répondre aux actions suivantes :

- **READ(*pageId*)** – URL : `index.php?title=pageId`
 - Affiche le contenu formaté de la page *pageId* – si la page n'existe pas, sa création est proposée -
- **EDIT(*pageId*)** – URL : `index.php?title=pageId&action=edit`
 - déclenche l'édition de la page *pageId* et sa création si elle n'existe pas. Le contenu textuel de la page est présenté à l'utilisateur dans une zone d'édition. L'utilisateur peut sauvegarder ses modifications ou abandonner son édition. L'abandon de l'édition conduit à afficher la page dans l'état dans lequel elle se trouvait avant édition.
- **SAVE(*pageId*)** : URL : `index.php?title=pageId&action=save`
 - Sauvegarde les modifications réalisées lors de la phase d'édition. Le contenu modifié devient le nouveau contenu de la page. La nouvelle page est affichée.
- **SEARCH(*string*)** : URL `index.php?title=search&q=string`
 - recherche les pages dont le nom contient « *string* » et affiche la liste des noms des pages répondant à la demande sous la forme de liens. Si la liste est vide, propose la création de la page de nom « *string* ».

Formatage et affichage :

La syntaxe pour le formatage des pages est celles de MediaWiki. Les éléments de base sont rappelés en annexe. L'affichage formaté des pages doit respecter les descriptions fournies.

- Les liens doivent être affichés **en bleu**
- les liens vers des pages non créées doivent être affichés **en rouge**

Interface

L'interface doit comprendre 3 zones :

- une zone dans laquelle est affichée le nom du wiki et le nom de la page courante,
- une zone de contenu utilisée pour afficher une page ou éditer une page,
- une zone d'action permettant de déclencher des actions sur la page courante et permettant de déclencher une recherche.

Sécurité

- Le code source directement inséré dans le texte wiki d'une page ne doit pas être interprété.
- Le wiki doit résister aux attaques élémentaires par passage de valeurs dans l'url (injection).

A rendre :

Votre projet doit être disponible sur une des machines de l'équipe. Il doit être accessible de la fin de l'épreuve web à la fin du concours. Le wiki DOIT OBLIGATOIREMENT CONTENIR :

- **les 2 pages correspondant aux 2 fichiers de test fournis SANS AUCUNE MODIFICATION. Ces pages doivent être nommées « MAIN » et « PageListes ».**
- **Une page nommée « PageFonctions », accessible depuis la page « MAIN » avec le lien prévu à cet effet, dont le contenu doit être une description des fonctionnalités réalisées dans votre projet, classées en 2 catégories :**
 - **les fonctions minimales : décrivez quelles fonctions et éléments de la syntaxe sont supportés,**
 - **les fonctions étendues : décrivez ce qui a été réalisé en plus, et indiquez toutes les données nécessaires à leur test (login, pass, nom de fichiers),**

Ces 3 pages doivent obligatoirement être présentes sur votre serveur final qui sera évalué. Elles serviront de point de départ aux testeurs.

La qualité de la page « PageFonctions » : honnêteté du contenu, précision des descriptions, présence des informations nécessaire au test ... fera partie de l'évaluation.

Fonctionnalités étendues

Ces fonctionnalités ne sont à réaliser que lorsque les fonctions minimales sont opérationnelles et testées. Vous pouvez les choisir dans l'ordre que vous voulez. Elles sont présentées dans l'ordre croissant des difficultés, et donc des points qu'elles rapportent. Elles sont indépendantes les unes des autres, sauf les 2 questions concernant les utilisateurs.

1. Prévisualisation en cours d'édition : lors de l'édition d'une page, l'utilisateur peut prévisualiser la page formatée, et décider de la sauvegarder en l'état ou de reprendre l'édition.
2. Gestion des conflits d'édition : lorsque un utilisateur demande l'édition d'une page qui est *déjà en cours d'édition*, un message avertissant l'utilisateur est affiché et une demande de confirmation est faite. Si l'utilisateur confirme sa volonté de modifier la page, il rentre en mode édition et édite normalement, avec le risque d'écraser les modifications faites en même temps par l'autre utilisateur (ou de perdre les siennes).
3. Images dans les pages : étendre le moteur de wiki pour ajouter la possibilité d'insérer des images dans les pages wiki. Les fichiers images doivent être téléchargés sur le serveur : il faut prévoir une interface pour cela. On inclut une image dans une page avec la syntaxe :
 - `[[Image:nomFichier|texte alternatif]]`
 - exemple : `[[Image:maFleur.jpg|Tournesol]]`
4. Gestion d'utilisateurs et de droits d'édition : réserver le droit d'édition à des utilisateurs enregistrés. Un utilisateur enregistré peut s'identifier auprès du wiki et peut alors éditer les pages. Un utilisateur non identifié peut parcourir le wiki et visualiser le source des pages mais pas le modifier. Le wiki doit être étendu pour permettre aux utilisateurs de s'identifier. Un utilisateur particulier dispose de la capacité à créer des utilisateurs.
5. Chaque utilisateur enregistré du wiki est représenté par une page du wiki qu'il est le seul à pouvoir éditer. Cette page représente son profil dans le wiki. Il peut modifier son mot de passe dans cette page.
6. Gestion de l'historique : lors de la sauvegarde des modifications sur une page, une nouvelle version de la page contenant le nouveau texte est créée et devient la version courante. L'action READ affiche par défaut le contenu de la version courante. Elle peut aussi recevoir un paramètre supplémentaire indiquant un identifiant d'une version particulière à afficher. Une nouvelle action est introduite : HISTORY(PageId) qui affiche la liste anté-chronologique (de la plus récente à la plus ancienne) des versions d'une page. Pour chaque version, la date de création est indiquée, et un lien permet d'afficher le contenu formaté de cette version.

ANNEXE

Syntaxe MiniWiki

(tiré de <http://www.mediawiki.org/wiki/Help:Formatting/fr>)

| Description | Vous tapez | Vous obtenez |
|-------------------------------|---|--|
| | <i>s'applique n'importe où</i> | |
| Texte en italique | <code>''italique''</code> | <i>italique</i> |
| Texte en gras | <code>'''gras'''</code> | gras |
| Gras et italique | <code>''''gras et italique''''</code> | <i>gras et italique</i> |
| Lien interne | <code>[[Page]]</code> | Page |
| Lien interne nommé | <code>[[Page texte différent]]</code> | texte différent |
| Lien externe | <code>http://www.mediawiki.org</code> | http://www.mediawiki.org |
| | <i>seulement en début de ligne</i> | |
| Titres de différentes tailles | <code>==Niveau 1==</code> <code>===Niveau 2===</code> <code>====Niveau 3====</code> | Niveau 1 Niveau 2 Niveau 3 |
| Ligne horizontale | <code>----</code> | <hr/> |
| Liste à puces | <code>* un</code> <code>* deux</code> <code>* trois</code> | <ul style="list-style-type: none">• un• deux• trois |
| Liste numérotée | <code># un</code> <code># deux</code> <code># trois</code> | <ol style="list-style-type: none">1. un2. deux3. trois |

Note importante :

- Il n'est pas demandé de supporter des balises imbriquées
 - exemple : `=== '' texte '' ===`
- Une balise de fermeture oubliée est automatiquement ajoutée en fin de ligne
 - exemple : `blahblah '''texte gras blahblahblah`
est interprété : `blahblah '''texte gras blahblahblah'''`

ANNEXE

Affichage des pages exemples

Affichage correspondant au source contenu dans le fichier : Main.txt

Main

MiniWiki : un wiki tout MiMi

Il est *mini*, il est **wiki** mon *miniWiki*

MiniWiki, mais il fait le maximum !

MiniWiki est un mini moteur de wiki créé à l'occasion de l'épreuve web des 24h des IUT Informatique 2012, organisée à Montreuil les 6 et 7 avril 2012.

Les étudiants issus de différents IUTs de France, par équipe de 4 à 6, ont produit ce remarquable outil en 8h de travail !

liens :

Ce lien doit être rouge : [PageInfo](#)

Ce lien doit être bleu : [PageListes](#)

Le lien vers le site des 24h : <http://iut-informatique.fr/24hinfo/> 

Le lien vers la page de description du projet, doit être rouge lors des premiers tests puis bleu lors du rendu final : [PageFonctions](#)

Affichage correspondant au source contenu dans le fichier : PageListes.txt

PageListes

Affichage de listes en MiniWiki

liste à puces

- item 1 : premier élément de la liste
- item 2 : deuxième élément, avec un lien vers la page [Main](#)
- item 3 : 3eme *élément*,
- item 4 : 4eme **item**

liste numérotée

1. premier élément de la liste
2. deuxième élément, avec un lien vers la page [Main](#)
3. 3eme *élément*,
4. 4eme **item**