**Premier mandat : Wiki**

Les procédures stockées sont à venir dans le cadre du cours des bases de données

Un wiki permet l'affichage et l'édition d'une série d'articles. Un article est défini par :

* Un titre,
* Un contenu,
* Une date de modification,
* Un numéro de révision
* L’id d'un contributeur. Le contributeur est la dernière personne à avoir modifié l'article.

Dans un premier temps votre application ne supportera pas l’authentification ; vous utiliserez la valeur 1 pour l'id du contributeur.

**Spécifications fonctionnelles**

L'affichage est divisé en 3 zones :

* Un menu,
* Une table des matières,
* Une zone de contenu.

Ces trois zones sont toujours présentes.

* Un clic sur un article de la table des matières affiche cet article.
* La table des matières est affichée en ordre alphabétique.

**Page d'accueil**

La zone de contenu affiche :

* Un message de bienvenue,
* Une zone de texte et bouton permettant de consulter l'article dont le titre apparaît dans la zone de texte.

**Affichage d’un article**

La zone de contenu affiche :

* L'article (le titre, le contenu formaté, la date de modification, l'id du contributeur, le numéro de révision)
* Un lien de modification.

Dans le cas d'un article inexistant, un message indique que l'article n'existe pas et un lien permet de l'ajouter.

**Ajout d’un article**

La zone de contenu affiche les champs suivants :

* Une zone de texte contenant le titre à ajouter et un text area pour le contenu (l'utilisateur peut entrer des balises html).
* Un lien permet d'annuler l'ajout et de revenir à la page d'accueil.
* Un bouton affiche un aperçu de l'article en dessous de la zone d'édition.
* Un bouton enregistre les modifications et retourne à l'affichage de l'article.

**Modification d’un article**

La zone de contenu affiche l'article en modification :

* Une zone de texte pour le titre
* Un text area pour le contenu non formaté (les balises html sont affichées).
* Un lien permet d'annuler la modification et de revenir à l'affichage de l'article.
* Un bouton affiche un aperçu de l'article en dessous de la zone d'édition.
* Un bouton enregistre les modifications et retourne à l'affichage de l'article.
* Un bouton permet de supprimer l'article.

**Suppression d'un article**

L'article est affiché et une confirmation est demandée. Un bouton confirme la suppression. Un lien annule la suppression et retourne à l'affichage de l'article.

**Routage des requêtes**

Sans modifier le routage des requêtes par le fichier *RouteConfig.cs*, faites-en sorte que votre application réponde aux requêtes suivantes :

**/** ou **/home** ou **home/index** : affiche la page d’accueil ;

**home/index/<titre de l’article>** : affiche l'article;

**home/index/<titre inexistant>**: invitation à créer l’article;

**home/modifier/titre:** modifie l'article;

**home/ajouter/titre**: ajoute un article;

**home/supprimer/titre**: supprime un article.

**Spécifications techniques Couche métier**

Une classe **Article** dans l’espace de noms **Models/Biz** (et dossier du même nom) permet de gérer les articles.



Le numéro de révision commence à 1 et est incrémenté à toutes les modifications. Dans un premier temps l'IdContributeur sera défini à 1.

**Couche d'accès aux données**

L’application accède à la base de données par ADO via une couche d’accès aux données constituée d’une seule classe dans l’espace de noms Models/DAL (et dossier du même nom).



**Find()** retourne null si l’article n’est pas trouvé.

**Add()**, **Update()** et **Delete()** retournent le nombre d’enregistrements modifiés dans la bd.

**GetTitres()** et **GetArticles** retournent une liste vide s’il n’y a pas d’article.

Pour accéder à la base de données, la classe Articles utilise des **procédures stockées. Celles-ci sont déjà présentes dans la base de données.**

La chaîne de connexion est stockée dans le fichier Web.Config

**Marche à suivre**

Récupérez le projet Wiki initial et déposez-le sur votre dépôt GIT.

1. La première étape consiste à développer la classe d'accès aux données. Voici comment chaque équipe va procéder :

* Récupérer la base de données et intégrez-la à votre projet ;
* Codez votre méthode en indiquant en commentaire vos noms;
* Testez votre méthode à l'aide du contrôleur DAL;
* Lorsque votre méthode est au point, faites un update du fichier sur GIT pour récupérer les méthodes des autres;
* Testez à nouveau votre méthode **et celles des autres**;
* Comparez votre code et celui des autres et refactorisez au besoin. Si vous pensez que le code des autres pourrait être améliorer,
* Discutez-en avec eux. Faites un commit et annoncez que votre méthode est codée

1. Vous êtes prêts maintenant à développer votre Wiki. Commencez par implanter votre routage. Chaque requête définie dans la section Routage des requêtes doit appeler la bonne méthode d'action dans votre contrôleur.
2. Implantez les fonctionnalités en commençant par la page d'accueil et l'affichage d'un article. Terminez par la table des matières.
3. Lorsque tout est fonctionnel, faites-en sorte que l'utilisateur puisse entrer des balises Html.