



Cégep de Saint-Hyacinthe  
Département d'informatique

## Développement d'application Web (420-GEC-HY)

### Travail Pratique sur une application Web (Phase 2 – Ajout de fonctionnalités)

Préparé par  
**Martin Lalancette**  
bureau D 2330A

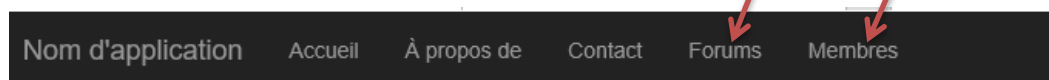
## DESCRIPTION

But :	Intégrer les bases de données et les niveaux d'accès à une application Web.
Objectifs :	<ul style="list-style-type: none"><li>• Produire la documentation propre à une application Web.</li><li>• Gérer les bases de données via les modèles (Code First – <i>Entity framework</i>)</li><li>• Gérer des rôles et des membres (<i>Identity</i>)</li></ul>
Durée :	6 h
Pondération :	sur 10 ( <b>Seul ou en équipe de deux</b> )
Remise :	À la semaine 10.
Contenu général :	<ul style="list-style-type: none"><li>• Remettre votre copie du travail via LÉA</li><li>• Conserver une copie de sécurité. Il est de <b>votre responsabilité</b> de conserver une copie de sécurité dans l'éventualité où la lecture des données serait impossible. Cette copie doit <b><u>être disponible sur demande</u></b>.</li></ul>
Notes	

### Introduction et contexte



La compagnie GEC Informatique à vérifier et valider le travail effectué lors de la phase 1. Elle a présenté le tout au client qui alors se dit satisfait de ce début de conception de l'application. Le client voit plus loin et souhaite ajouter de nouvelles fonctionnalités dans la prochaine phase. GEC, dont la devise est **satisfaction du client**, vous charge d'ajouter ces fonctionnalités qui sont décrites dans la prochaine section de ce document. Elle tient à souligner la qualité de votre travail et souhaite que vous continuiez dans cette voie. Voici donc la description de chacune des fonctionnalités demandées. Ajout au menu :



### Fonctionnalité #1 : Un forum.

Le client souhaite permettre aux utilisateurs d'avoir un espace de discussion publique afin qu'ils puissent échanger entre eux, leurs opinions. Voici les exigences du client :

1. Il faut être inscrit comme membre pour pouvoir donner son opinion.
2. Un rôle « modérateur » devra être défini. Le membre ayant ce rôle pourra :
  - a. Ajouter/Modifier/Retirer des sujets
  - b. Effacer des messages qu'il juge offensants
  - c. Exclure un membre des forums
3. La **page d'accueil du Forum** doit afficher la liste des sujets avec les informations minimales suivantes pour chaque sujet: Titre du sujet, description du sujet, Nombre de publications (Posts), l'auteur qui a créé le sujet avec sa date de création, le dernier usager avec la date publiée (ou temps écoulé). Exemple :

**Voici la liste des sujets traités dans notre forum :**

Nombre de messages répondus.

**Ajouter un sujet**

Titre	Description	Auteur	Dernier	#	Actions
<a href="#">Les meilleurs terrains!</a> <div>Ici, l'utilisateur clique pour accéder aux messages.</div>	Nous voulons connaître les meilleurs terrains que vous avez fréquentés dans la pratique de votre sport.	Mike13 2015-09-15	Lucie234 2015-09-27	23	<a href="#">Modifier</a> <a href="#">Supprimer</a>
...	...	...	...	...	


Ce qui est en **rouge** est **disponible seulement pour le modérateur** (et plus → niveau d'accès). Modifier un sujet consiste à pouvoir changer le titre et la description.

4. La **page d'un sujet** doit afficher l'ensemble des messages rattachés à un sujet et doit contenir minimalement les informations suivantes : auteur et son image, date de création du message, le message. **L'ordre du plus récent au plus vieux message est important.** Exemple :

**Sujet :** [Les meilleurs terrains!](#)

**Description :** [Nous voulons connaître les meilleurs terrains que vous avez fréquentés dans la pratique de votre sport.](#)

**Ajouter un message**

Auteur	Message	Action
Mike13  2015-09-28 11:39	Bonjour,  Nous aimerions connaître vos préférences en matière de terrains. Écrivez-nous en grand nombre!  Mike123	<a href="#">Supprimer</a>
Lucie234...	...	...

Ce qui est en **rouge** est **disponible seulement pour le modérateur** (et plus → niveau d'accès). Ce qui **en vert** est **disponible pour les membres inscrits** (et plus).

5. Une fois publié, un message ne peut pas être modifié ou effacé par un utilisateur.
6. Créer une vue pour chaque action sur un message: afficher la liste, l'ajout et la suppression.
7. Créer une vue pour chaque action sur un sujet: afficher la liste, l'ajout, modification et suppression.

Vous devez embellir ce prototype en utilisant diverses composantes (CSS, Bootstrap, etc.).

### Fonctionnalité #2 : Gestionnaire de membres

Cette fonctionnalité consiste à gérer les informations associées aux membres du site. Voici les détails :

1. Créer un modèle nommé **AspNetUsersInfoSup** qui sera chargé d'emmagasiner les informations supplémentaires (image du profile, etc.) d'un utilisateur.
2. Définir les rôles de base suivant dans la table **AspNetRoles** (coder via Migrations).

Rôle	Description
Administrateur	Accès à tout.
Utilisateur	Accès au minimum. Membre inscrit (par défaut).
Modérateur	Gestion du Forum (ajout de sujets, suppression de messages)
ExclusionForum	Le membre ayant ce rôle, se voit exclu du forum. Ne peut y aider sous aucune condition.
<i>Autre</i>	<i>Ajouter d'autres rôles en lien avec votre projet.</i>

3. Ajouter un contrôleur **Membre**.
4. Associer un ou plusieurs rôles à un usager via la table **AspNetUserRoles**. Ajouter les vues (Lister, Ajouter, Modifier, Supprimer) pour un usager. Seulement un Administrateur pourra y accéder.
5. Définir les accès aux contrôleurs impliquant des rôles.
6. Faire afficher/masquer les éléments touchant les rôles dans les vues impliquées.

D'autres informations vous seront communiquées lors de la séquence sur *Identity*.

Gestion du projet :

- Pour s'assurer du bon contrôle du projet, vous devez **remplir chaque semaine une feuille de temps** à remettre auprès de la compagnie (votre enseignant).

<b>Barème d'évaluation pour cette phase</b>
---

Fonctionnalité #1 – Page d'accueil du Forum .....	/1.0
Fonctionnalité #1 – Page d'un sujet (liste des messages) .....	/1.0
Fonctionnalité #1 – Gestion des pages du modérateur .....	/1.0
Fonctionnalité #1 – Gestion des messages faits par l'utilisateur .....	/1.0
Fonctionnalité #2 – Création/gestion de la tableAspNetUsersInfoSup.....	/0.5
Fonctionnalité #2 – Établissement des rôles (table AspNetUserRoles) .....	/0.5
Fonctionnalité #2 – Gestion des membres et rôles (Lister, Ajouter, Modifier, Supprimer) .....	/1.0
Fonctionnalité #2 – Application des restrictions d'accès dans l'application.....	/1.0
Qualité générale (présentation, professionnalisme)** .....	/3.0
<b>Note totale*</b> .....	<b>/10.0</b>

\* Tout travail plagié en partie ou en totalité se verra attribuer une note totale de 0 %.

\*\* Concernant l'évaluation de ces étapes, l'enseignant peut demander une démonstration auprès de l'équipe.

**PDEA #1:** Lors d'activités d'évaluation sommative en classe ou hors classe (documentation, rapport de laboratoire, rapport de stage, examen), une **pénalité maximale de 10 %** peut être retranchée de la note finale de ladite évaluation (le barème étant de **0,5%/erreur incluant les fautes répétitives**). Pour les commentaires dans les programmes informatiques, le barème est de **0,5% par faute**. Pour les interfaces utilisateurs, le barème est de **1% par faute**.

**PDEA #3 :** Toute évaluation sommative remise **après la date d'échéance fixée** se voit **attribuer la note zéro**.