

**Rapport Final**  
**Équipe 1**

Par  
Luc-Antoine Girardin  
Nicolas Lavoie Drapeau  
Lionnel Lemogo  
Alexandre Mathon-Roy  
William Tchoudi

présenté à  
Stéphane Vaucher

Université de Montréal  
Développement et Maintenance  
cours IFT3912  
18 avril 2014

## Architecture

Au départ, il était planifié que le projet soit construit avec un client et un serveur avec un pont qui gère la communication entre les deux. L'architecture a peu changé en bout de ligne, mais certains changements ont été effectués un peu partout dans le logiciel.

Le pont était supposé être constitué d'un seul Handler, qui allait chercher les permissions dans différentes classes du pont. Cependant, dans l'architecture finale, il a été convenu que le pont contienne plusieurs Handler, sans classe pour gérer les permissions. Chaque classe gère donc ses propres permissions. La raison du changement est simple : c'était plus facile de travailler en équipe avec une telle architecture. Le pont est plus simple d'utilisation pour les autres développeurs.

Les classes du client s'occupent des affichages et d'envoyer au serveur, via le pont, les informations nécessaires à l'exécution d'une tâche particulière. Le pont s'assure qu'il n'y a pas d'erreurs et envoie les informations au serveur dans le bon format. Le serveur s'occupe alors d'exécuter la tâche, potentiellement en accédant à la base de données, et répond au client.

Afin de gérer les requêtes, les requêtes héritant de `__RedirectHandler__` furent remplacées par des Handler héritant de Servlet. Ce changement a permis de simplifier l'utilisation de contexte et offre une manière de gérer les sessions et de faire des redirections plus facilement. Cependant, la redirection des servlet n'a pas été implémentée pour éviter aux utilisateurs d'avoir à fournir de nouveau certaines informations lors d'erreurs. À la place d'afficher une page web d'erreur, des fenêtres pop-up ont été implémentées pour avertir l'utilisateur de l'erreur qui a causé des problèmes.

L'authentification des usagers est faite lors des requêtes qui demandent l'accès ou qui modifient des informations privilégiées. Essentiellement, le changement est que les cookies gardent maintenant le mot de passe de l'utilisateur afin qu'un utilisateur ne puisse pas obtenir l'accès à n'importe quel compte en mettant le nom d'utilisateur dans le cookie. Il est donc plus difficile, mais pas impossible, d'obtenir des accès privilégiés.

Finalement, pour que l'âge de l'utilisateur se mette automatiquement à jour, c'est la date de naissance qui est mémorisée dans la base de données. L'âge est ensuite calculé au niveau du serveur.

## Rapport de tests

Les tests ont beaucoup été exécutés par assertion. Pour vérifier si les valeurs sont les bonnes à tout moment, une condition est vérifiée. Si la valeur booléenne est fausse, le programme s'arrête et les causes du problème peuvent être identifiées.

La base de données a été testée en créant une classe TestDB qui créait une base de données temporaire. Ensuite, elle testait une fonction de la base de données et vérifiait si le

résultat était celui espéré. Les fonctions pouvaient toutes être testées par cette classe. Les assertions ont aussi été utilisées pour la base de données.

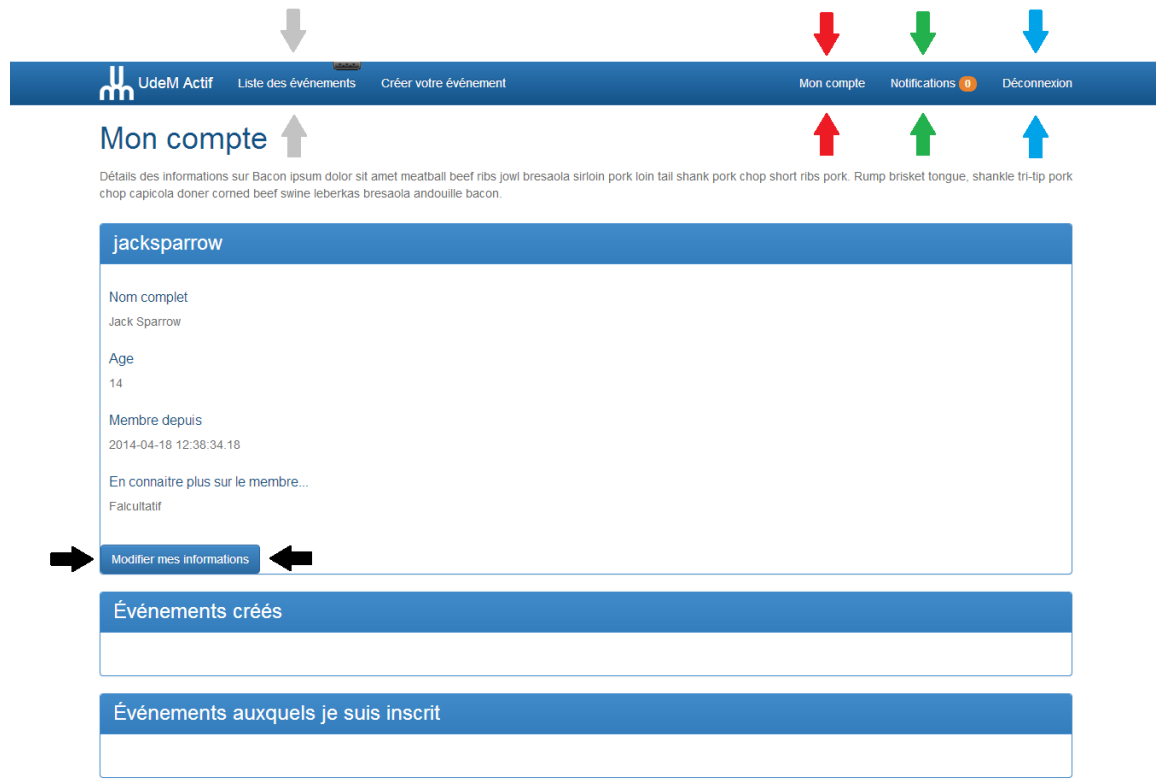
Des tests de qualité ont été effectués en chargeant le site web et en testant la réaction du site face à des fonctions, surtout des fonctions qui peuvent inclure des entrées critiques (mauvais format pour un attribut entier par exemple). L'analyse des réactions du site a pu éviter certains problèmes.

## Manuel d'utilisation

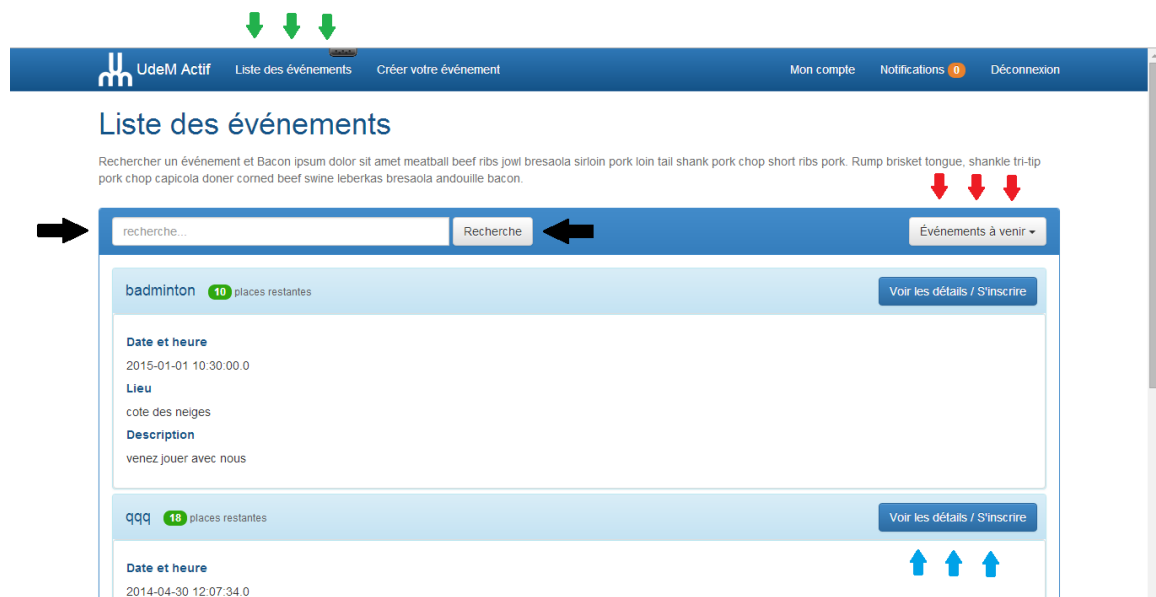
La page d'accueil ressemble à l'image ci-dessous.



Pour créer un compte usager, il suffit de cliquer sur « Enregistrez-vous » en haut sur la barre bleue à droite. Après avoir rempli les informations d'utilisateur, il faut cliquer sur « Créer mon compte » pour finaliser la demande.



La page du nouvel utilisateur s'affichera. Pour modifier les informations du compte, il suffit de cliquer sur « Modifier mes informations » (flèches noires). L'utilisateur peut toujours revenir sur sa page en cliquant sur « Mon compte » (flèches rouges). Il y a un bouton « Déconnexion » (flèches bleues) pour se déconnecter du site. Pour visualiser ses notifications, il suffit de cliquer sur « Notifications » (flèches vertes). Toujours sur la même page, en cliquant sur « Liste des événements » (flèches grises), une liste d'évènement à venir s'affiche.




Lorsqu'on arrive sur la page de liste d'évènements comme sur la dernière figure de la page précédente, on peut alors utiliser la barre de recherche (flèches noires) pour chercher un évènement par mots clés. En cliquant sur « Évènements à venir » (flèches rouges), un menu déroulant s'affiche. L'utilisateur peut alors choisir d'afficher une liste d'évènements passés, ou une liste d'évènements annulés. Pour créer un évènement, il faut cliquer sur « Créez votre évènement » (flèches vertes).

Comme pour créer un utilisateur, en remplissant les champs pour ensuite cliquer sur « Ajouter l'évènement », l'évènement sera créé et sera affiché dans la liste des évènements à venir. Pour s'inscrire à un évènement, il suffit de cliquer sur « Voir les détails / S'inscrire » (flèches bleue de la dernière image) de l'évènement dans la liste auquel l'utilisateur veut s'inscrire.

The screenshot shows the UdeM Activ interface. At the top is a navigation bar with the UdeM logo, 'UdeM Actif', and links for 'Liste des événements', 'Créer votre événement', 'Mon compte', 'Notifications' (with a red badge), and 'Déconnexion'. Below this is the 'Détails de l'événement' section. It features a title 'badminton' and statistics: 'Par: pat', 'Personnes inscrites: 4', and 'Places restantes: 6'. The event details include 'Quand?' (2015-01-01 10:30:00.0), 'À quel endroit?' (cote des neiges), and 'Détails sur l'événement' (venez jouer avec nous). A blue 'S'inscrire!' button is highlighted with two red arrows. Below the details is a section 'Exprimez-vous sur cet événement' with a text input field, an 'Envoyer' button, and a list of comments. One comment is visible: 'pouet' from user 'ccc' on 2014-04-18 12:10:55.776. Two green arrows point to the comment input and the 'Envoyer' button.

Il y a alors un bouton « S'inscrire! » (flèches rouges). Un utilisateur peut également laisser un commentaire et visualiser les commentaires d'autres utilisateurs en bas de la page (flèches vertes). On peut en tout temps cliquer sur « Mon compte » (flèches noires) pour visualiser les évènements créés par l'utilisateur en question et ceux auxquels il s'est inscrit.

Il peut alors accéder aux évènements qu'il a créés en bas de sa page d'utilisateur pour les modifier ou les supprimer en cliquant sur « Modifier l'évènement » (bouton bleu sur l'image de la page suivante) et « Supprimer l'évènement » (bouton rouge) respectivement.

 UdeM Actif

Liste des événementsCréer votre événement

Mon compteNotifications 0Déconnexion

## Détails de l'événement

Trouvez les informations sur Bacon ipsum dolor sit amet meatball beef ribs jowl bresaola sirloin pork loin tail shank pork chop short ribs pork. Rump brisket tongue, shankle tri-tip pork chop capicola doner corned beef swine leberkas bresaola andouille bacon.

Un évènement quelconque

Par: lucaPersonnes inscrites: 0Places restantes: 18

Quand?

2014-04-30 13:31:59.0

À quel endroit?

Quelque part

Détails sur l'événement

On va faire des choses comprends-tu?

Modifier l'événement

Supprimer l'événement

Exprimez-vous sur cet événement

Écrivez un commentaire

## Problèmes connus et limitations

Le site contient certains problèmes de sécurité pour les utilisateurs. Notamment, lors de l'enregistrement d'un nouvel utilisateur, il n'y a pas de confirmation de mot de passe. S'il fait une faute lors de la création de celui-ci, il ne saura pas pourquoi il n'est pas capable de se connecter ultérieurement. Aussi, s'il essaie de se connecter et qu'il se trompe de mot de passe trop souvent, son compte ne bloquera jamais, Un individu peut essayer de deviner le mot de passe d'un individu jusqu'à l'infini. Il y a très peu de restriction par rapport aux informations de l'utilisateur. Il peut même inscrire une date de naissance dans le futur. Aussi, parfois les lettres accentuées ne s'affichent pas correctement.

## Contribution

Alexandre a travaillé beaucoup sur le front end. Il a travaillé sur les classes HTML, CSS et javascript. Il a également travaillé sur l'intégration, les requêtes Ajax et la création des Templates avec Mustache. Il a aussi aidé à faire du débogage et des tests.

William a construit la base de données et a fait les tests sur celle-ci. Il a aidé à la conception du logiciel et la création du prototype. Il a aidé à créer les premiers rapports entre autre en faisant le diagramme de Gantt et la structure du WBS. Il a aidé à coder l'architecture et un peu le pont.

Luc-Antoine a rédigé la majorité des textes pour les rapports. Il a travaillé sur l'implémentation de l'architecture et le codage du pont. Il a aidé à faire des tests et du débogage.

Nicolas a aidé à développer l'architecture du projet. Il a aidé aussi à la rédaction des premiers rapports. Il a travaillé fort à déboguer GitHub lors de la fusion des branches. Il a également aidé à faire des tests unitaires et déboguer certaines fonctionnalités.

Lionel a implémenté la majorité des Handlers du pont. Il a aussi aidé à faire les Templates avec Mustache.

## **Discussion**

Dans le déroulement du projet, la distribution des tâches s'est fait vite puisque l'expérience des programmeurs marchait bien avec la structure du programme. En effet, un seul des programmeurs était vraiment à l'aise de programmer une base de données, et un seul était à l'aise avec la programmation web pour les fichiers HTML. Les autres ont partagé le reste du travail. Les membres de l'équipe étaient assidu à leurs tâches donc les remises se sont fait sans véritable problème.

Par contre, la communication a été difficile en dehors du peu de rencontre d'équipe qu'il y a eu. En effet, la communication se faisait par courriel. Cependant, certains ne regardaient pas leur courriel régulièrement, d'autre ne pouvait pas venir aux réunions pour cause d'autres projet dans d'autres cours, etc. Donc la coordination de l'équipe en a souffert. Par ailleurs, GitHub a été grande cause de problèmes car deux branches ont été créées et les membres de l'équipe ne travaillaient pas tous sur la même branche. Lorsqu'il a fallu fusionner les branches, GitHub a été très capricieux et la fusion a créée des conflits dans le code. Beaucoup de temps a été perdu pour corriger les problèmes.

L'organisation a changé dans la mesure où Jetty étant nouveau pour tous les membres du groupe, il a fallu mettre plus de ressources dans l'implémentation du serveur. Nous avons donc attribué les interfaces à Alexandre vu son expérience tandis que les autres ont pu se concentrer sur comprendre et implémenter Jetty.

Si le projet avait à être recommencé sans contraintes, Jetty ne serait pas utilisé. Il a causé des problèmes dans le sens où personne ne savait comment l'implémenter alors que d'autres outils étaient connus et auraient pu être utilisés. La manière d'obtenir certaines informations dans les classes pourrait être uniformisée pour faciliter l'entretien du code.

Le projet a donc su démontrer l'importance d'une bonne communication au sein d'un groupe de développement. En effet, avec une communication plus fluide et facile, des problèmes de communication auraient pu être évités. Par exemple, l'architecture a changé en cours de projet entre autre puisque le pont avait été commencé d'une façon sans consensus et a du être recommencé après discussions. Une solution aurait d'avoir des réunions périodiques pour éviter l'organisation d'une rencontre chaque semaine. Trouver une disponibilité commune est plus facile lorsqu'on force les coéquipiers à ne pas organiser autre chose à un moment précis. Les réunions auraient permis de se mettre à jour régulièrement dans le développement et éviter certaine frustration comme lorsque quelqu'un touche au code sur lequel on travaillait.

Une autre leçon apprise fut que si un collègue a des difficultés, il est plus approprié de l'aider ou lui montrer comment faire la tâche en question que de le faire à sa place. En effet, la distribution des tâches se déséquilibre et la personne ne pourra pas plus le faire plus tard s'il y a lieu. Si on lui montre comment faire, son expertise augmente et l'équipe peut encore plus compter sur la personne.

Chacun des membres de l'équipe a augmenté sa capacité à travailler en équipe pour le développement d'un logiciel.