

UNIVERSITÉ DU QUÉBEC À MONTRÉAL

RAPPORT DE SPRINT FINAL

ALEXANDRE DUFOUR  
DUFA23059001

ALEXANDRE H. BOURDEAU  
HAMA12128907

JORDAN GAUTHIER  
GAUJ25089201

PHILIPPE SOUCY  
SOUP17039601

TRAVAIL PRÉSENTÉ À  
ABDOULAYE BANIRÉ DIALLO PH. D  
DANS LE CADRE DU COURS  
INM5001 ATELIER

LE 24 AVRIL 2020

## Table des matières

<b>Identification du projet et de la personne responsable .....</b>	<b>3</b>
<b>Description du mandat .....</b>	<b>3</b>
Différentes modifications et ententes conclues.....	3
<b>Description détaillée des activités planifiées et réalisées à cette étape .....</b>	<b>4</b>
Détails sur les fonctionnalités complétées partiellement .....	4
Analyse, conception, programmation, validation .....	5
Outils et technologies utilisées.....	5
<b>Identification des solutions apportées aux problèmes rencontrés dans le sprint 1 .....</b>	<b>6</b>
Problème #1.....	6
Problème #2.....	6
Problème #3.....	6
Problème #4.....	6
Problème #5.....	7
Problème #6.....	7
Problème #7.....	7
Problème #8.....	8
<b>Problèmes généraux rencontrés .....</b>	<b>8</b>
Problème #9 – Alexandre H. Bourdeau .....	8
Problème #10 – Alexandre H. Bourdeau .....	9
Problème #11 – Philippe Soucy .....	9
Problème #12 – Philippe Soucy .....	9
Problème #13 – Alex Dufour-Couture .....	9
<b>Évaluation du nombre d’heures consacrées au projet en entier .....</b>	<b>10</b>
<b>Évaluation du produit fourni .....</b>	<b>10</b>

## Identification du projet et de la personne responsable

Le projet se nomme « Plateforme Web événementielle et interactive ». L'application est développée par le groupe « La marque sans nom » qui est composé de Jordan Gauthier, Alex Dufour-Couture, Philippe Soucy, Alexandre H. Bourdeau (responsable).

## Description du mandat

Le mandat consiste à concevoir, développer et programmer une plateforme Web interactive de création, d'inscription et de partage d'événements entre des organisations et des particuliers. Les utilisateurs de la plateforme pourront échanger et collaborer dans le but d'organiser collectivement des événements. Des organisations telles que des centres sportifs ou de loisirs pourront utiliser la plateforme pour la location de leurs infrastructures et équipements, et ainsi alimenter la banque de ressources événementielles collective.

La plateforme interactive doit permettre à une personne de s'inscrire à la plateforme. Une fois inscrite, plusieurs options s'offrent à elle : consulter la banque d'événements et s'inscrire comme participant, organiser un événement autour d'un lieu proposé par une entreprise ou organiser un événement autour d'un autre lieu. Les entreprises, quant à elles, peuvent offrir des infrastructures qui peuvent être louées par les usagers et aussi proposer des événements.

Les usagers de la plateforme peuvent communiquer entre eux à l'aide d'un système de messagerie interne. Les participants d'un événement peuvent communiquer entre eux à l'aide d'un babillard propre à chaque événement.

Le client est M. Diallo, Abdoulaye Baniré Ph. D.

## Différentes modifications et ententes conclues

Compte tenu des heures perdues dues à la COVID-19, nous nous sommes entendus pour réduire les exigences de certaines fonctionnalités en plus d'en abandonner quelques-unes.

Nous avons donc retiré ces fonctionnalités du sprint 3 :

- Chat entre utilisateurs,
- Affichage d'événements similaires,
- Service de commerce (panier) en ligne afin de louer équipements ou infrastructures,
- Tableaux statistiques

## Description détaillée des activités planifiées et réalisées à cette étape

#	Fonctionnalité	Statut
	Conceptualisation du schéma relationnel	Terminé
	Injection des données	Terminé
8	Implémentation d'un gestionnaire d'événements (désinscription, téléversement d'une image, modification, suppression)	Partiellement complété
10	Implémentation d'une messagerie privée entre utilisateurs (front-end)	Partiellement complété
11	Modifier les informations/préférences de comptes utilisateur (y compris le téléversement d'un avatar)	Partiellement complété
	Babillard	Terminé
13	Ajout et affichage de services d'une entreprise	Terminé
15	Implémentation d'une barre de recherche d'événements (par ville, sport, activité, etc.)	Terminé
16	Affichage d'historique d'événements	Terminé

### Détails sur les fonctionnalités complétées partiellement

#### *Implémentation d'un gestionnaire d'événements/Comptes utilisateur - téléversement d'une image*

Le support back-end d'un téléversement d'image sur le serveur a été implémenté. Cependant, nous avons eu de la difficulté à faire le lien avec le front-end. Une page existe qui permet de téléverser un avatar, mais nous aurions voulu ajouter cette fonction directement dans les options du profil utilisateur. Aussi, nous n'avons pas eu le temps d'implémenter l'équivalent en ce qui concerne l'image couverture des événements. Mais nous estimons que ces fonctionnalités ne prendraient pas trop de temps à implémenter si nous avions eu quelques heures de plus.

#### *Implémentation d'une messagerie privée entre les utilisateurs*

Une partie de la messagerie est avancée, cependant il reste des fonctionnalités à implémenter comme la possibilité de répondre aux messages reçus. C'était un défi que nous voulions accomplir pour pouvoir en apprendre davantage, cependant il faut peut-être avouer que nous aurions dû utiliser une librairie nous permettant d'intégrer directement un système de messagerie préexistant. La tâche était assez compliquée.

### Analyse, conception, programmation, validation

Un travail d'analyse a été effectué afin d'apprendre le téléversement de fichier et sa sécurité, en ASP.NET, incluant Entity Framework.

Un travail de programmation a été effectué afin de terminer les modifications, suppression d'événement et d'entreprise.

Plusieurs validations sur les formulaires ont été effectuées, la documentation et les correctifs ont été faits.

### Outils et technologies utilisées

Nous avons utilisé les outils et technologies suivants.

#### *Environnement de développement*

- Visual Studio 2019,
- Visual Studio Code,
- Docker,
- Gestionnaire de source Github
- Azure Data Studio,
- SQL Server Management,
- SQL Server

#### *Front-end*

- Bootstrap,
- JQuery,
- FontAwesome (pour les icônes),
- Tempus Dominus DateTimePicker (<https://tempusdominus.github.io/bootstrap-4/>)

#### *Langage et frameworks*

- C#,
- Microsoft ASP.NET Core,
- Entity Framework,
- Design pattern MVC (Model-View-Controller)

## Identification des solutions apportées aux problèmes rencontrés dans le sprint 1

### Problème #1

Connexion à la base de données problématique, soit des issues de migrations et de mises à jour de base de données.

Solution : Nous avons sauvegardé plusieurs configurations différentes afin d'éliminer les différences entre nos systèmes d'opération. Le serveur est également individuel, avec seulement le fichier migration commun à chaque développeur. La remise à zéro de la base de données est souvent effectuée pendant le développement, puisque la structure de celle-ci change au fur à mesure.

### Problème #2

Aucun rôle assigné à un nouvel utilisateur.

Solution : Nous avons dû ajouter le champ à l'entité de l'ApplicationUser (objet qui désigne l'utilisateur connecté), puis ajouter le code nécessaire à l'affectation du rôle par défaut lors de l'inscription de l'utilisateur.

### Problème #3

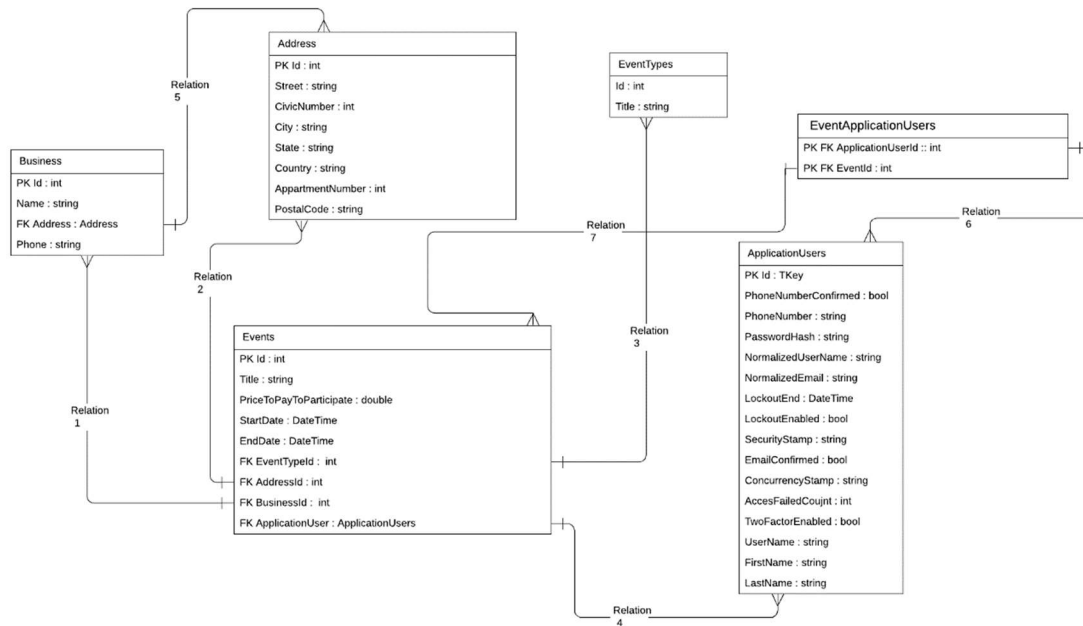
Styles non cohérents partout (méthodes en français, d'autres en anglais).

Solution : Nous avons décidé de renommer les méthodes et de poursuivre les méthodes en anglais, puisque l'échafaudage de code (Entity Framework et Identity) est automatiquement en anglais.

### Problème #4

Problèmes de relations entre les différentes tables dans la base de données.

Solution : Création d'un dessin UML afin d'aider à la conception des relations.  
Sprint#3: Changement des méthodes faisant la relation entre les tables, afin d'inclure plusieurs tables.



### Problème #5

Problème de lancement de programme selon les versions et dépendances requises.

Solution : Nous avons arrêté les mises à jour automatiques et avons commencé le projet sans utiliser la dernière version. L'installation a été faite en début de projet et nous avons donc pu nous aider en temps réel.

### Problème #6

Problème d'intégration des gabarits des pages à la branche du projet.

Solution : Les pages gabarits statiques ont été intégrées manuellement et retravaillées pour ne pas interférer avec les différents formats. Ceci était seulement en début de projet afin de laisser la chance à chaque membre d'avancer leurs parties respectives et établir l'environnement de travail indépendamment.

### Problème #7

Complexité à se connecter à Docker sur Linux dû à une absence d'interface graphique. Cela a rajouté un peu de temps d'apprentissage.

Solution :

**Philippe Soucy**

Comme j'avais d'abord un tutoriel sur comment paramétrer Docker par l'entremise de l'interface graphique disponible sur Mac, j'ai dû chercher sur le web comment faire les mêmes configurations, mais en ligne de commande. Le grand problème, et c'est revenu tout au long de la confection du projet, c'est que j'ai eu plusieurs erreurs d'autorisations d'accès. J'ai appris qu'il est très important d'utiliser `sudo` strictement quand c'est nécessaire, car si on exécute une commande qui change les droits d'accès à un dossier en étant super-utilisateur, on ne pourra plus construire le projet et le rouler. J'ai eu donc plusieurs fois à changer des groupes et des droits d'accès pour réussir à lancer le site web.

**Problème #8**

Nous avons eu de la difficulté à nous connecter à la base de données et par conséquent avons dû travailler une bonne partie du temps en « local ».

Solution :

**Alexandre Bourdeau**

J'ai dû faire plusieurs recherches sur le web afin de trouver une solution à ce problème. Finalement, j'ai trouvé un article sur StackOverflow qui expliquait la façon de créer un nouvel utilisateur avec les droits administrateurs. J'ai pu ensuite modifier le fichier de configuration de connexion à la BD pour y mettre le nouvel utilisateur créé.

**Problèmes généraux rencontrés**

**Problème #9 – Alexandre H. Bourdeau**

Nous avons eu de la difficulté à implémenter un composant de type "DateTimePicker" qui permet à un utilisateur de la plateforme de sélectionner une date et une heure dans une interface graphique, à l'aide de la souris. Ce composant est pratique pour la sélection de la date et l'heure ainsi que sa validation.

Un exemple de problème rencontré à ce sujet :

Un formulaire de modification de données doit sauvegarder les données lorsque le formulaire est soumis. Lorsqu'une date était choisie avec le composant, la sauvegarde ne s'effectuait pas.

Solution :

J'ai remarqué que dans certains cas il ne faut pas utiliser toutes les fonctionnalités du framework ASP.NET Core MVC. Par exemple, ce framework permet de générer facilement des contrôles de type "input" à l'aide de modèles. Lorsqu'on procède de cette façon, la propriété "value" est générée automatiquement avec la valeur dans la base de données. Cependant, ça ne fonctionne pas bien avec le composant "DateTimePicker" car celui-ci ne change pas la valeur de la propriété "value". Au lieu de cela, il stocke la valeur dans une variable Javascript ("data-target"). Lorsque le formulaire était soumis, la valeur envoyée au serveur était celle dans la propriété "value". C'est pour cela que la sauvegarde ne s'effectuait pas.



### Problème #10 – Alexandre H. Bourdeau

Dans ASP.NET, nous pouvons utiliser le tilde (~) pour faire référence au répertoire “wwwroot” qui se trouve à la racine du projet. Cela nous permet d'afficher une image facilement dans n'importe quel répertoire du site sans avoir à nous soucier du chemin relatif. Cependant, pour une raison que j'ignore, ça ne fonctionnait pas dans tous les cas.

Solution :

J'ai dû faire une recherche sur le web pour trouver une solution dans les forums de Microsoft pour ASP.NET. Il y existe une fonction qui permet de convertir un chemin relatif en chemin absolu. Je l'ai utilisé et tout a fonctionné.

### Problème #11 – Philippe Soucy

Pour la suppression d'entreprise, lorsqu'on clique sur supprimer, notre site affiche une page de confirmation. Lorsqu'on supprimait un événement, il n'y avait pas de confirmation. Le problème était que l'on n'avait pas le temps de créer une page de confirmation spécifique aux événements, en raison de la COVID-19 et du report de plusieurs fonctionnalités. Il y avait donc là une incohérence sur les différentes pages de notre site. De plus, ne pas demander de confirmation de suppression peut avoir des conséquences très peu désirables du côté client.

Solution :

J'ai créé un message de confirmation rétractable pour que l'utilisateur assure son choix.

### Problème #12 – Philippe Soucy

Compiler le site et charger la base de données dans un conteneur local était un défi pour une personne comme moi qui n'avait pas de cours de base de données ou de programmation web. J'ai donc dû apprendre .NET à la dure. Un problème qui était récurrent était la construction du projet qui échouait en produisant une erreur indiquant “value key null”. Ce problème est en extension du problème #1, mais il mérite une attention particulière puisqu'il était récurrent.

Solution :

La solution a été de supprimer, à plusieurs reprises, les migrations qui contenaient l'état de la base de données, et d'en refaire une nouvelle, avant de charger celles-ci dans le conteneur.

### Problème #13 – Alex Dufour-Couture

Beaucoup de problèmes associés avec Entity Framework Core, et plus spécifiquement avec le modèle relationnel de la base de données. Entre autres, un des problèmes est que Entity Framework possède une structure prédéterminée et nous devons respecter son modèle structurel.

Solution: Nous avons dû recommencer la structure de base de données plusieurs fois et modifier les entités.

## Évaluation du nombre d'heures consacrées au projet en entier

Au départ, nous avons planifié consacrer 135 heures de travail par personne, pour un total de 540 heures.

À la suite des événements liés à la COVID-19, nous avons soustrait 18 heures de travail par personne, pour un total de 72 heures. Ces heures sont reflétées dans les fonctionnalités retirées du projet (voir plus haut).

À la fin du sprint 2, nous avons donc prévu de 473 heures au projet.

Le projet étant complété, nous avons consacré un total de 483,75 heures, soit un surplus de 10,75 heures.

## Évaluation du produit fourni

Le produit fourni est fonctionnel et comporte les fonctionnalités demandées selon ce qu'il a été possible de réaliser dans le temps imparti. Le temps de réalisation a dépassé les heures estimées de peu. Le produit a cependant plusieurs fonctionnalités qui auraient pu être approfondies si nous avions eu plus de temps.

L'aspect sécurité et contrôle de qualité ont dû être faits un peu par chaque membre, ce qui mettrait la plateforme plus à risque si nous avions à la déployer. Nous n'avons pas mis en production le produit. Nous l'avons seulement développé localement. Cela nous amènerait à prévoir une bonne période de temps supplémentaire si nous avions à réaliser cette seconde phase du projet.

Le produit fait partie d'un projet académique qui nous a fait comprendre tout le temps nécessaire à l'accomplissement de certaines tâches dans un contexte d'entreprise. Nous pouvons par exemple mieux comprendre l'importance d'avoir une équipe dédiée au contrôle de qualité. Plusieurs problèmes à régler s'accumulent au fur et à mesure que le projet grandit, et donc une personne dédiée assure un suivi plus strict.

L'utilisation de Framework et ses outils nous ont permis d'offrir plusieurs fonctionnalités communes en peu de temps. Nous avons su implémenter de manière efficace les librairies utilisées, ce qui nous a permis de nous concentrer sur le développement de fonctionnalités plus spécifiques à notre produit.

Le produit est à jour en termes de technologies, et la migration vers la prochaine version de ASP.NET ne sera pas nécessaire avant un bout. La maintenance du logiciel et la migration vers un API sont également facilement concevables grâce au gestionnaire de base de données, Entity Framework et le design pattern MVC.