

## Rapport de sprint 1

### 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.

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

#### Analyse, conception, programmation, validation

Un travail d'analyse a été effectué afin d'apprendre la librairie *C# ASP.NET* et ses frameworks enveloppant le design pattern *MVC (Models, Views, Controllers)*.

Un travail d'analyse a été effectué afin d'apprendre la façon d'héberger une image *Microsoft SQL Server* sur un *Docker container* (à l'aide du logiciel *Docker*).

Le schéma relationnel de la base de données a été créé (voir annexe 1).

Le schéma relationnel des classes qui interagissent avec la base de données a été créé (voir annexe 2).

Les maquettes de plusieurs pages web (accueil, contact, utilisateurs, etc.) ont été conçues.

## Tâches prévues dans Sprint 1

PREMIER SPRINT	
Connexion a la DB, et installation Docker / Microsoft	Terminé
Data seeding	En cours
Template accueil, event-view, event-list, profile-view	Terminé
Creation de model Entreprise	Terminé
Template du site - user-list, profile-edit	Terminé
Schema design pattern	Terminé
Travail sur Controlleur : Account, admin,error,event	Terminé
Account management settings. Front end/ back, Asp Razor Pages implentation	Terminé
Système d'authentification / Enregistrement	Terminé
Administration des roles des utilisateurs	Terminé
Conceptualiser le schema relationnel de la Base de donnée	En cours

## Tâches en cours

La conceptualisation du schéma relationnel sera en cours tout au long du projet jusqu'à sa livraison.

L'injection des données sera également faite tout au long de l'avancement du projet.

## Outils et technologies utilisées

Nous avons utilisé les outils suivants :

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

## Période d'apprentissage

Une période d'apprentissage a été nécessaire afin de nous familiariser avec *C#, le framework .NET Core, ASP.NET Core MVC/Razor, Bootstrap, HTML, CSS, Shiny Dashboard*. L'apprentissage des outils est également inclus.

Nous avons planifié une moyenne de 25 heures par personne pour l'apprentissage et nous familiariser avec les outils et la documentation. À ce jour, nous en sommes à 17 heures réelles par personne.

## Identification des problèmes rencontrés

Nous avons eu plusieurs problèmes, en voici quelques-uns :

- Connexion à la base de données problématique, soit des issues de migrations et de mises à jour de base de données,
- Il n'y a aucun rôle assigné à un nouvel utilisateur (sera corrigé bientôt),
- Styles non cohérents partout (méthodes en français, d'autres en anglais),
- Problèmes de relations entre les différentes tables dans la base de données,
- Problème de lancement de programme selon les versions et dépendances requises,
- Problème d'intégration des templates des pages à la branche du projet. Cela nous a ralenti d'une journée (8 heures) mais sera complété pour la livraison du premier sprint,
- Complexité à se connecter à *Docker* sur *Linux* dû à une absence d'interface graphique. Cela a rajouté un peu de temps d'apprentissage,
- 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 »

## Réajustement de la planification

Aucun réajustement ne sera nécessaire puisque nous sommes à jour sur l'échéance du projet.

Voici les tâches prévues dans le sprint 2 :

1	Appliquer un style uniforme à la plateforme
2	Administration des rôles des utilisateurs
3	Pages administration et gestion par GUI
4	Pages gestion d'entreprises (Create,Edit)
5	Pages gestion d'événement (Create,Edit,Delete)
6	Messageries entre utilisateur
7	Babillards de commentaires
8	Rédiger le rapport de sprint 2

## Évaluation du nombre d'heures consacrées à ce jour

Nous avons planifié consacrer 135 heures par personne pour un total de 540 heures. Nous avons consacré environ 275 heures au total.

## Estimation du nombre d'heures restant pour la livraison finale

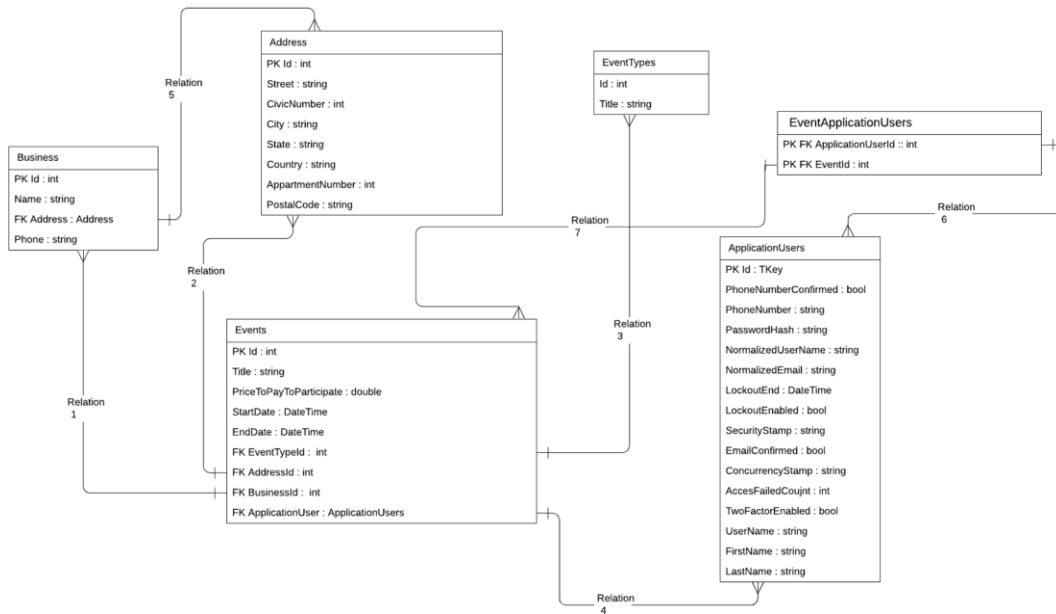
Il reste environ 265 heures à consacrer au projet.

## Leçons apprises

- L'installation de l'environnement joue un grand rôle dans le temps,
- Il est important de *push* et *pull* souvent pour éviter *les merge conflicts*,
- Il est important de bien communiquer aux membres de l'équipe la partie sur laquelle on travaille et ce qui est planifié pour ne pas travailler « pour rien »

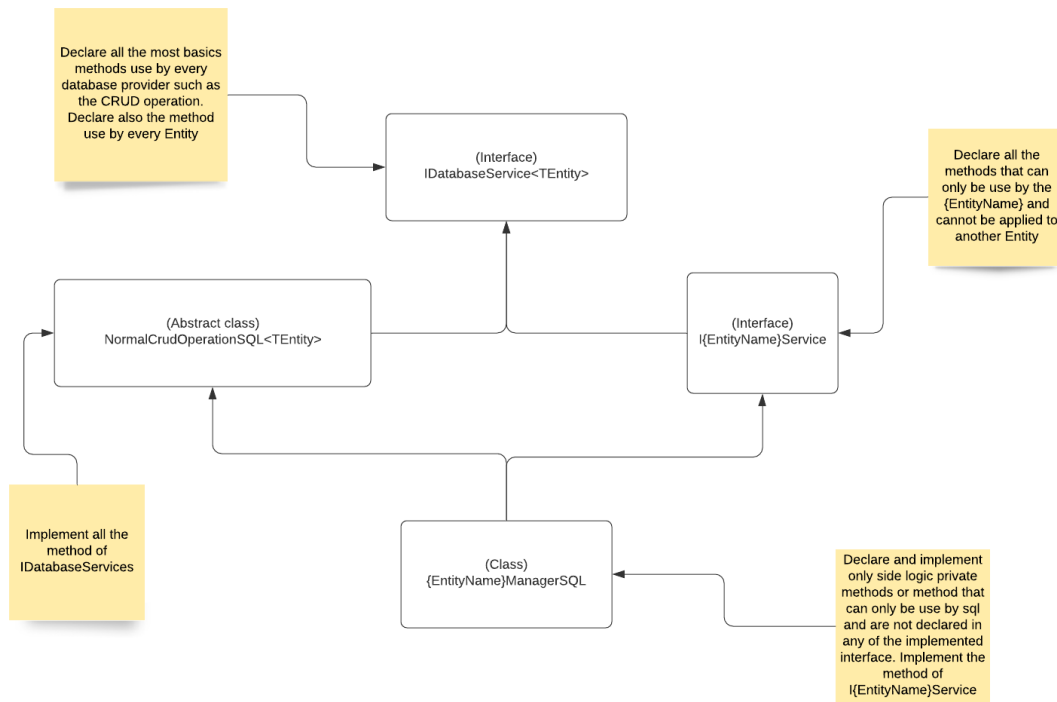


## Annexe 1



*Le schéma relationnel de la base de données.*

## Annexe 2



*Le schéma relationnel des classes qui interagissent avec la base de données.*