

Projet web 2016 – Rapport

Vincent BOCHET, Romain DROUIN, Elish REBSTOCK, Guillaume SÉZILLE

10/05/2016

Composition du groupe

Notre groupe est composé des membres suivants :

- Vincent BOCHET (4.2)
- Romain DROUIN (3.1)
- Eliah REBSTOCK (4.1)
- Guillaume SEZILLE (4.1)

Problématique et motivation

Notre projet est un projet libre, néanmoins inspiré du projet d'application de ranking pour babyfoot proposé par votre équipe. Ainsi, nous avons également choisi de travailler sur une application de classement, mais plutôt pour des jeux de société compétitifs. Le site est donc issu d'un besoin de compétition sur ces jeux, mais également d'un besoin d'interaction sociale à travers ces compétitions.

La raison principale de ce choix est presque tous les membres de notre groupe sont eux-mêmes des joueurs de jeux de société au sein de l'ENSIIE et que nous avons trouvé cela motivant de pouvoir travailler sur un projet nous concernant, et pouvant même être réutilisé par la suite une fois le projet terminé.

Approche mise en place

Afin de réaliser ce projet, il nous a semblé clair que l'architecture MVC (Modèle – Vue – Contrôleur) était la plus adaptée. En effet, celle-ci permet de réutiliser plusieurs fois les mêmes choses dans divers contextes : ainsi, les méthodes du modèle (permettant les accès à la base de données) peuvent être réutilisées dans différentes vues, au travers du contrôleur. De même, certaines méthodes d'un même contrôleur peuvent être utilisées dans plusieurs vues. Outre l'avantage de ne pas réécrire plusieurs fois le même code, la structure MVC permet aussi une maintenance beaucoup plus aisée, puisqu'on peut ajouter des fonctionnalités de façon modulaire en ajoutant un nouveau modèle, un contrôleur et de nouvelles vues associées (ou simplement en étendant les modules déjà existants).

Nous avons donc, dans un premier temps, réfléchi sur les diverses fonctionnalités à implémenter sur notre application et les cas d'usage possibles. Nous avons de cela tiré un certain nombre de pages web, permettant des actions spécifiques. Dans le même temps, nous nous sommes lancés dans la conception d'une première version de la base de données, en fonction des besoins identifiés. En parallèle, nous avons cherché comment réaliser un site web avec l'architecture MVC, et avons créé quelques pages afin de bien nous familiariser avec le fonctionnement.

Nous avons utilisé Git durant toute la durée du projet afin de pouvoir travailler séparément, chacun sur un bout du projet, en créant des branches pour chaque nouvelle fonctionnalité. Il suffisait ensuite de merge les branches afin d'obtenir un tout cohérent, ce qui fut facilité par l'architecture MVC. Par ailleurs, afin de pouvoir communiquer à l'intérieur du groupe à n'importe quel moment, nous avons utilisé Slack, un service de messagerie pour les équipes.

Problématiques rencontrées (techniques et méthodes) et solutions apportées

Au cours de notre développement, nous nous sommes heurtés à plusieurs problèmes, et ce à différents niveaux. D'un point de vue méthodologique, nous faisons l'erreur de fonder nos fichiers (vue/contrôleur) sur la base de données, ce qui nous a été indiqué lors de la deuxième séance de projet. Cela avait tendance à nous brider un peu sur le développement de certaines fonctionnalités, et une fois ce problème identifié, nous avons pu nous libérer de cette contrainte, modifiant la structure de la base et rajoutant des modules autant que nécessaire. Par ailleurs, les nombreux changements sur la base par chacun des membres de l'équipe ont entraîné souvent des divergences entre les différentes branches, et ont donc demandé de nombreuses régénérations de bases dans chacune des bases personnelles de développement. Cela aurait pu être corrigé en travaillant directement avec une base sur un serveur commun de développement.

Il y a également eu des débats internes au groupe sur l'algorithme de classement à utiliser. En effet, au début du projet, l'algorithme ELO n'était pas une évidence pour tout le monde, et un simple système d'ajout/retrait de points fixes semblait suffire. Mais, en expliquant les principaux intérêts de l'algorithme (prendre en compte le niveau des joueurs pour attribuer la quantité de points) le groupe fut finalement convaincu de la nécessité d'utiliser cet algorithme. Cependant, le fait de devoir adapter l'algorithme à plusieurs types de jeu demandait des efforts supplémentaires, que certains membres n'étaient pas prêts à prendre. Mais Eliah Rebstock, qui avait proposé l'algorithme après quelques recherches sur le sujet, accepta de prendre charge de celui-ci.

Enfin, un manque d'objectifs au début du projet nous a menés à ne pas beaucoup avancer les premiers temps, car aucune liste précise des choses à faire n'était établie. Nous avons tenté de corriger le problème à l'aide des issues sur Git, mais c'est finalement la réalisation d'une liste de TODO qui nous a réellement permis d'avancer de façon productive. De cette façon, chacun pouvait se lancer dans le développement d'un point de la liste, et une fois ce point terminé, pouvait se lancer dans un autre sans nécessité de se concerter avec les autres membres, puisque ceux-ci étaient occupés sur d'autres tâches déjà identifiées.

Au niveau technique maintenant, nous avons rencontré d'autres soucis. Par exemple, un bug qui ne réalisait pas les bonnes redirections sans raison apparente. Il s'est avéré que le souci venait d'un manque d'accolades dans le code, ainsi que de la nécessité de cesser l'exécution du reste du fichier après la redirection à l'aide de la fonction `die()` ; de PHP. Parmi les soucis techniques, nous pouvons également penser au fait qu'aucun d'entre nous n'avait utilisé MVC en pratique dans un contexte web, qu'aucun d'entre nous ne maîtrisait JavaScript, ni le responsive design en CSS, et pour palier à ces soucis, il nous a fallu avoir une démarche d'autodidacte, ce qui a ralenti le développement.

Répartition des tâches

Une répartition des tâches s'est mise en place dès les travaux préparatoires sur le projet. Tout d'abord, nous avons énuméré ensemble les différentes fonctionnalités et usages que nous souhaitions avoir sur le site et si cela correspondait bien au principe du site. L'équipe a donc dégagé les fonctionnalités suivantes : l'inscription au site, l'ajout de partie, l'affichage de profil,

l'affichage de jeu et du classement associé, l'affichage d'un classement global.

Pendant une première phase du projet, Guillaume Sézille a mis en place une première version de la base de données, Vincent Bochet a travaillé sur la mise en place d'une structure MVC utilisable par tous les membres du groupe, tandis que Romain Drouin réfléchissait à de nouvelles fonctionnalités. En parallèle Eliah Rebstock a fait des maquettes pour le site et mis en place des pages tests afin d'implémenter le design qui allait devenir celui du site. Certaines parties comme les headers et footers seront reprises dans le design final, tout comme certaines pages converties en vues.

Une fois que la structure MVC a été mis en place, il fut possible de commencer à travailler sur les différents modèles du site. Ainsi, Eliah Rebstock s'est focalisé sur l'ajout de partie, la mise en place d'un algorithme ELO amélioré capable de gérer les différents types de jeu disponibles et la page de classement global. Vincent Bochet a mis en place l'authentification ainsi que les pages d'accueil, de contact, la page de profil, le menu, la migration de la base de données sous PostgreSQL (au lieu de MySQL). Romain Drouin a, lui, pris en charge la gestion des images pour les jeux et les membres, l'édition de profil et l'interface d'administration dans sa totalité.