



MOTEUR DE JEU DE STRATÉGIE WEB

PROJET TI5

Dossier Bilan

Auteurs :

Ilyas BOUTEBAL
Maxime COLIN
Adrian GAUDEBERT
Youness HAMRI
Van Duc NGUYEN

Client :

Pierre-Antoine CHAMPIN

21 février 2011

Table des matières

1	Contexte	2
2	Livrables	3
2.1	Code source	3
2.2	Documentation	4
3	Organisation	5
3.1	Outils et méthodes	5
3.2	Répartition des tâches	5
3.3	Concordance avec les prévisions	5
4	Bilan	6

Chapitre 1

Contexte

Le projet Fightly a pour objectif la création, dans un premier temps, d'un moteur de jeu Web orienté stratégie au tour par tour, puis dans un deuxième temps de promouvoir ce moteur, notamment en créant des jeux l'utilisant. Ce projet est libre, il devra donc vivre et évoluer par et pour sa communauté.

Le but de notre travail, dans le cadre de l'UE TI5 de notre formation, a donc été de lancer ce projet, en développant une première version minimale mais fonctionnelle. Ce document présente l'avancement du projet Fightly à la fin de cette première étape, les objectifs atteints, la façon dont le projet s'est déroulé, et notre bilan sur l'ensemble de l'UE TI5.

Chapitre 2

Livrables

Le cahier des charges du projet indiquait que nous devions livrer, au terme du projet, trois éléments différents :

À la fin du projet, on fournira au client les trois éléments suivants :

- Le code source complet de l'application développée;
- Une documentation technique, à destination des développeurs qui utiliseront le moteur, mais également à l'intention des développeurs qui contribueront par la suite au projet, celui-ci étant libre;
- Une démonstration technique du moteur.

2.1 Code source

Le projet étant libre, l'intégralité du code source est accessible en ligne, via le site de github.com : <https://github.com/AdrianGaudebert/flightly-game-engine>.

En l'état actuel, le code de l'application contient également le code de la démonstration technique : nous fournissons avec le code du moteur de jeu un ensemble de fichiers de configuration qui permettent de simuler un jeu très simple (avec gestion des tours, déplacements et attaques d'unités), ainsi qu'un ensemble de fonctions et d'images permettant de gérer l'affichage de cette démonstration.

Le moteur en tant que tel n'est pas encore réellement générique, puisque certains aspects sont peu paramétrables, ou trop interdépendants. Cependant, la structure a été pensée pour permettre l'extensibilité, le travail à effectuer restera donc minime pour arriver à une version plus générique du moteur.

Nous n'avons pas atteint l'objectif d'origine du projet, qui était de créer un moteur utilisable, générique et extensible. Ce résultat s'explique par la complexité du projet, mais également par des choix qui ont été faits en accord avec le client. Nous avons en effet rencontré, comme dans tout projet, des problèmes imprévus qu'il nous a fallu résoudre, et qui ont retardé notre avancement. Etant donné le temps relativement réduit du développement, nous avons choisi de simplifier certaines problématiques, et de nous concentrer sur des problèmes liés directement aux mécaniques du moteur plutôt qu'à des problèmes de généricité. Si nous avons donc une version démontrable techniquement, elle n'est pas totalement extensible.

2.2 Documentation

La documentation finale du projet devra être découpée en deux parties : une à destination des développeurs internes au projet, et une à destination des utilisateurs du moteur (les créateurs de jeux). Nous avons fait le choix, pour l'instant, de ne pas séparer ces deux documentations. Nous ne fournissons donc, au terme de ce projet pour l'université, qu'une seule documentation technique, disponible sur le wiki technique du projet : <http://fightly-dev.lqbs.fr/wiki/>.

De plus, le code source de notre application est intégralement documenté en suivant les normes JSDoc. La documentation ainsi générée est accessible aux adresses <http://fightly.com/api/server/> (côté Serveur) et <http://fightly.com/api/client/> (côté Client) et devra être intégrée au reste de la documentation du projet.

Chapitre 3

Organisation

3.1 Outils et méthodes

Afin de gérer ce projet, nous avons principalement utilisé le forum du projet Fightly (<http://fightly-dev.lqbs.fr/forum/>), par le biais duquel nous avons communiqué sur les réunions et les informations importantes. Une grande partie des échanges s'est cependant faite en réunion.

Pour le développement et une partie de la documentation, nous avons utilisé un serveur Git. Nous avons d'abord utilisé SourceForge, pour migrer en cours de projet vers Github, plus stable et dont les outils fournis étaient suffisants.

Nous nous sommes efforcés de faire du Test Driven Development tout au long du projet, afin de toujours disposer de procédures de test effectives. Nous avons cependant constaté que les tests unitaires n'étaient pas facilement adaptables à un moteur de jeu, dont les données sont souvent très complexes. Nos tests ne sont donc pas complets, au sens qu'ils ne testent pas l'intégralité des fonctionnalités et des comportements de notre moteur.

3.2 Répartition des tâches

La répartition initiale des tâches (voir Cahier des charges) a globalement été respectée. Chaque membre du groupe a travaillé sur un ou deux domaines particuliers du moteur, et s'y est tenu. Nous avons cependant travaillé par itérations, en suivant une méthode agile, en mettant en place tout d'abord la structure de l'application, puis en ajoutant régulièrement des fonctionnalités. De cette façon, nous pouvions tester régulièrement notre application, et vérifier que la base de notre code marchait avant d'ajouter une nouvelle couche.

3.3 Concordance avec les prévisions

Nous n'avons globalement pas respecté les délais dont nous avons convenu dans notre cahier des charges. Les raisons sont multiples : comme dit précédemment, nous avons rencontré des imprévus dans le développement. Certains d'entre nous ont également eu du mal à cumuler ce projet et les nombreux TPs et autres projets. Afin de rattraper un peu de notre retard, nous avons choisi de simplifier notre étape d'analyse, et de commencer plus rapidement le développement. Ce choix s'est avéré utile, étant donné nos cycles de développement itératifs qui permettent de passer moins de temps sur la conception. Enfin, nos estimations pour le développement étaient grandement surestimées : nous avons mis presque le double du temps prévu initialement.

Chapitre 4

Bilan

Cette expérience s'est, de notre point de vue, particulièrement bien déroulée. Nous avons su faire face à nos obstacles, et surtout nous avons su réaliser une application qui, si elle est loin d'être finie, est au moins fonctionnelle. Nous avons beaucoup travaillé ensemble, ce qui nous a permis d'être dynamiques et attentifs au travail des autres.

Ce projet nous a apporté une expérience de développement sur une application complète, de la conception au développement. L'échelle du projet est bien plus grande que ce que l'on fait dans les TPs ou projets habituels, l'expérience acquise l'est donc aussi puisqu'on doit répondre à des problématiques qu'on ne rencontre pas autrement.

Nous estimons cependant que ce projet est trop peu valorisé dans la formation. Tout d'abord, le coefficient de la note nous paraît trop faible par rapport au temps passé sur le projet. Nous avons conscience qu'il est difficile pour vous de vérifier le temps de travail et l'implication de chacun, mais nous pensons qu'un plus fort poids de cette UE dans la formation conviendrait mieux. Nous regrettons également le manque de temps dédié au projet, nous aurions apprécié avoir une journée complète par semaine consacrée uniquement à ce projet, pour se rapprocher un peu plus d'un véritable projet d'entreprise, où les horaires de travail sont imposés.