



Projet Génie Logiciel

## **Bilan de Gestion D'équipe et de Projet**

Equipe: Equipe GL23

Professeurs: **Patrick Reignier & Tarik Larja**

Date : 26 Janvier 2022

## 1 Introduction

Ces dernières semaines passées à travailler sur le projet GL n'étaient pas uniquement axées sur les aspects techniques du travail, mais elles contenaient divers autres aspects que nous devons prendre en compte, et différentes nouvelles compétences que nous devons acquérir et mettre et appliquer.. et cela rentre principalement dans le large panel de la Gestion d'Equipe et de Projet.

En effet, afin de garantir le bon déroulement du projet et pouvoir gérer un projet de cet envergure, on a dû appliquer plusieurs méthodes et techniques apprises dans le cours, afin de rester organisés et concentrés, et d'éviter les problèmes dus à une mauvaise gestion de notre équipe.

## 2 Organisation générale Et Outils Utilisés

Afin d'implémenter une gestion de projet assez rigoureuse, on s'est assuré d'utiliser plusieurs outils introduits dans le cours de Gestion de Projet. En effet, en relisant tous les slides et les supports de cours, on s'est intéressé à sélectionner les outils et méthodes qui nous semblaient nous aider le plus à maintenir notre organisation, tout en étant cohérents avec notre manière d'opérer.

### 2.1 La Charte D'équipe

La charte d'équipe était, comme on l'avait précisé lors des suivis, une partie essentielle dans l'organisation de notre travail, vu que ça nous avait aidé à définir les grandes lignes de notre organisation, ce qui nous a fait gagner beaucoup de temps sur le long terme.

Afin d'élaborer cette charte, on avait organisé toute une réunion où l'on avait révisé ensemble le cours et où on a essayé de sélectionner les outils qu'on voudra employer dans notre travail. Déterminés à sortir de notre zone de confort et repousser nos limites, nous avons décidé de découvrir réellement les Méthodes AGILE et d'investir plus de temps pour en découvrir le fonctionnement. Ainsi, en gardant à l'esprit qu'on échangera bien nos rôles durant la réalisation du projet afin que tout le monde ait une chance pour acquérir ces nouvelles compétences, nous avons réparti au début les rôles les plus importants :

- Le Product Owner (Alae) qui sera responsable de la définition des besoins clients.
- Les Scrum Masters (Nada & Zineb) qui seront responsables de l'implémentation du planning et de la méthodologie, et de leur maintien.

Ensuite, lors de la réalisation de la charte, on avait décidé qu'on préférerait comme équipe travailler en distanciel, tout en imposant des réunions journalières à 22h chaque soir vu que la plupart des membres de

l'équipe préfèrent travailler assez tard la nuit. Et ces réunions quotidiennes se sont avérées être l'une de nos meilleures décisions sur le plan organisationnel. En effet, elles ont fait un excellent travail pour nous tenir responsables de terminer les tâches auxquelles nous nous sommes engagés pour la journée, afin d'en discuter et de les présenter au groupe. C'était aussi une belle occasion de discuter littéralement de tout et de rien concernant le projet... lors de ces réunions, outre notre avancement, nous discutons généralement des choix importants d'implémentation, des difficultés rencontrées, des tâches restantes et de leur répartition... C'était vraiment un excellent moyen d'impliquer tout le monde dans toutes les parties du projet, et d'être simplement tous conscients de ce qui a été fait et de ce qui reste.

Par contre, il faut avouer, que bien que ce mode de travail était bien très pratique tout au long des premières semaines du projet, on a commencé à sentir ses inconvénients vers les derniers jours où l'on est devenu pressés par le temps, où les tâches devenaient beaucoup plus interdépendantes, et où il fallait communiquer un peu plus fréquemment et rapidement que par messages sur discord, ou même par appels individuels... Et c'est dans de moments pareils où on succombait et on s'assemblait dans l'école dans la mesure du possible afin de travailler au même endroit de manière plus fluide.

Il faut aussi avouer que tous nos choix faits lors de la réalisation de cete charte n'étaient pas judicieux. En effet, on avait choisi de se contenter de laisser chaque personne tester la partie qu'elle code pour plus de simplicité. Mais juste après le cours technique qui suivait, et qui portait sur les différentes méthodes de testing, on a découvert que ce n'était point pertinent, et qu'il fallait laisser des personnes dédiés que à cette tâche précise, afin qu'ils puissent tester tous les détails des spécifications en restant complètement neutre par rapport aux choix de conception.

## 2.2 Les Méthodes Agile (SCRUM):

C'est aussi lors de la réalisation de la charte qu'on avait bien compris la méthode Agile dans les détails, et qu'on a décidé de l'utiliser et d'appliquer tous les outils qui lui sont liés, surtout en s'appuyant sur toutes les fonctionnalités qu'offre le Gitlab.

Ainsi, on a réparti le projet en différents Sprint par ce qu'on voyait être des "blocs de sens" ou des "blocs de tâches". Chaque Sprint durait à peu près une semaine, et donc nos trois principaux sprints étaient :

- Hello World / Suivi 1
- Sans Objet / Rendu Intermédiaire
- Objet / Rendu Final

Chaque Sprint commence par une réunion de kick-off, où on définissait les principaux objectifs de notre travail, les tâches à faire et leur répartition... Et il se termine par une réunion de clôture pour débriefer de ce qui a été fait, pour mettre à jour le planning de réalisation tous ensemble, et pour parler de ce qui vient.

Ensuite, chacun de ces sprints a été bien sûr découpé en User Stories formalisées par des tâches, ayant chacune un responsable (ou plus), et un délai dans lequel elle est supposée être terminée. Ensuite ces tâches seront ensuite réparties selon leur statut (ouvertes/ En cours / Fermées) dans le tableau.

L'utilisation de ces tableaux, bien qu'on l'ait faite au début principalement pour le cours, s'est avérée très éclaircissante de notre avancement, de ce qu'on a fait et ce qui nous reste... Et vers la fin on a continué à le faire de manière très spontanée et presque intuitive même tellement c'est devenu une partie intrinsèque de notre manière d'opérer en tant que groupe.

On avoue par contre ne pas avoir bien appréhendé le concept de ces tâches au début de notre travail vu qu'on définissait d'assez grosses tâches du genre "Partie B - Sans Objet", ou "Sans Objet - Génération de Code" et "Sans Objet - Testing". Et bien que ce soient tous des tâches qu'on devait bien réaliser, mais c'étaient vraiment de gros blocs qui prenaient des jours à être finis, et donc à être cochés. Ainsi, non seulement les colonnes "en cours" de notre tableau n'évoluaient que très lentement bien qu'on travaille très régulièrement, mais il y avait aussi le fait que la Burndown Chart ne reflétait point notre travail et nos efforts... elle restait juste une droite à peu près constante pendant les premiers jours du sprint. Et ce n'est que quelques jours avant le 2<sup>e</sup> sprint qu'on a découvert ce problème et tenté de le résoudre. Ainsi, à partir de ce suivi, nos tâches étaient beaucoup plus petites et pertinentes. On cite par exemple des tâches élaborées dans le sprint Objet : "Partie C - New", "Partie C - déclaration des champs", "Testing - Objet - Contextuel"... Ainsi, l'utilisation de la Burndown Chart comme outil pour évaluer notre avancement n'a été pertinent que depuis le 2<sup>e</sup> suivi, et principalement durant tout le 3<sup>e</sup> sprint.

On a aussi profité de l'opportunité que présentait gitlab pour associer à chaque tâche un poids donné, réfectif de l'effort de sa réalisation, afin de découvrir le "Planning Poker". En fait, ces temps passés à débattre quel poids associer à chacune des tâches étaient parmi les moments les plus drôles de nos réunions... ils étaient aussi importants pour évaluer l'importance et la difficulté de chacune des tâches avant de se lancer dans le bain.

## 3 Le Planning

Comme demandé dans les consignes du projet on a bien réalisé deux types de planning : Un de "prévision", et le second de "réalisation" concrète (tous deux accessibles sur le Git de notre projet).

La mise-à-jour du planning de réalisation ne se faisait pas de manière journalière, mais plutôt à peu près deux fois par sprint. On essayait de le rendre le plus précis possible, et le plus réfectif du temps réel passé dans chacune des tâches. On s'y référait aussi pour vérifier combien s'approchent les prochains délais afin de ne rien manquer, et de toujours être vigilants par rapport à ce qu'on devrait rendre à des dates précises.

## 4 Historique du Projet et Répartition des tâches

Au tout début du projet, on avait décidé d'être tous très impliqués dans toutes les tâches faites, et de tous lire les parties A et B afin que tout le monde ait une idée claire du fonctionnement du compilateur avant de se lancer dans l'hyperspécialisation des membres dans leurs tâches.

Au début on se doutait que ce soit une perte de temps, vu qu'on passait assez de temps dans la lecture du poly et des slides, mais ceci s'est avéré un exercice important pour le groupe, et même encouragé par le prof lors du suivi. En effet, non seulement on comprend tous comment fonctionne le compilateur du début à la fin, ce qui est une des plus grandes plus-values de ce projet, mais ça a aussi facilité l'entraide entre les membres du groupe, vu qu'on comprenait tous ce que faisaient les autres.

Ensuite, que ce soit dans la partie Sans Objet ou ensuite en Objet, on procédait de la même manière à chaque fois ; On travaille simultanément sur les parties A et B avant de passer tous à la partie C... En Parallèle, on assigne une personne au testing et une autre personne au travail sur l'extension.

Cette manière d'opérer était assez intuitive et nous a été très naturelle. Mais on a vite découvert qu'elle était loin d'être parfaite vu qu'on commençait assez tard le travail sur la partie C à chaque fois, alors que c'est l'une des parties qui nécessitent le plus de travail. Mais ça restait tout de même gérable vu qu'une fois qu'on la commence, on se mobilise tous pour la finir, et en général ça avance très vite.

Pour ce qui est de la logique du travail dans chacune des tâches, on s'assure d'abord de comprendre le travail qui devrait être fait, de faire la conception de comment on le fera, et de l'implémenter. Pendant ce temps, la personne responsable du testing aurait préparé toute une base de tests sur la partie, et en la lançant, on arrive à détecter les erreurs si elles existent, et on refait jusqu'à élimination de toutes les coquilles. On discutera ensuite ce qu'on a pu faire avec le reste du groupe dans la réunion journalière.

La répartition des tâches figurant sur les GANTT montrés s'est faite de manière très naturelle. En effet, comme on l'avait précisé lors du première et deuxième suivis, on avait laissé la répartition des missions entre les différents membres de l'équipe au libre choix. Et ceci s'est avéré être une très bonne décision, vu que, en choisissant ses propres tâches, chacun se sentait motivé pour les faire et s'engageait pour veiller sur leur complétude. Et en général, bien qu'il y ait souvent un esprit d'entraide et de communication entre les différents membres, on était tous autonomes et assez efficaces dans notre travail. Ceci a fait qu'on ne perdait pas de temps à revoir ce qui a déjà été implémenté par les autres membres, ni à refaire leur travail.

## 5 Remarques sur la Répartition des Tâches et de l'Organisation

En général, on a tous été extrêmement satisfaits par notre répartition des tâches, et chacun a été assez fier du travail qu'il a pu présenter et de l'autonomie et efficacité dont il a pu faire preuve. Mais ceci n'empêche

point qu'il y ait des failles dans cette répartition, et des choses qu'on aurait pu mieux faire.

En effet, outre le fait que nous aurions préféré si nous n'avions pas laissé la partie C à la fin à chaque fois, on aurait aussi aimé améliorer beaucoup plus notre approche au testing. On aurait pu par exemple assigner plus d'une personne à cette énorme tâche très importante pour avoir une base de tests beaucoup plus complète et exhaustive... Et ceci se remarque un peu plus sur nos batteries de tests des parties génération de code et Extension.

Aussi une des erreurs qu'on aurait vraiment aimé éviter était de mieux gérer la parallélisation des tâches dans la génération du code. En effet, vers la fin de la Partie Sans Objet, certains d'entre nous ont passé toute une nuit à changer radicalement la structure d'un ancien code sans en informer les autres membres du groupe jusqu'à un peu plus tard dans la journée. Ainsi, même si ces modifications étaient nécessaires, voir cruciales, les autres qui travaillaient en parallèle ont perdu assez de temps afin de tout ajuster à la nouvelle structure. Ceci aura pu être évité avec un peu plus de communication. Et c'est ce qu'on a tâché de faire dans la partie Objet... On s'appelait tout le temps, et on se notifiait mutuellement de toutes sortes de changements afin de mieux gérer le parallélisme dans notre travail. Et on a instantanément remarqué une amélioration dans la qualité de notre rendu.

Enfin, à la fin du projet, tout le monde est satisfait que nous ayons utilisé Git pour gérer notre travail de groupe, et plus précisément que nous n'ayons pas eu peur de travailler sur des branches différentes de celle du Master. C'était assez formateur et a pu révolutionner notre vision du travail en groupe.

## 6 Conclusion

Même si nous n'avions pas tout à fait saisi l'importance de ce projet au départ, nous nous sommes vite rendu compte à quel point il est crucial pour notre éducation et notre formation de futur ingénieurs en général. En fait, c'est l'un des projets scolaire avec la plus grande plus-value de toute notre formation. C'était une immersion directe dans la peau de vrais ingénieurs qui ont à comprendre un cahier de charge avec beaucoup de spécificités, et se méfier des plus petits détails en plus de ne rien assumer de ce que voudra le client.

Et en plus de tout l'apport scientifique duquel on a été doté grâce à ce projet, nous sommes également extrêmement reconnaissants de l'opportunité d'acquérir des compétences extra-techniques, surtout en matière de gestion de projet. On peut tous attester avec confiance qu'on utilisera désormais, assez intuitivement tous les outils qu'on a pu découvrir tout au long de ce mois!