

Grenoble INP – ENSIMAG, UGA
École Nationale Supérieure d'Informatique et de Mathématiques Appliquées

Projet Génie Logiciel

Bilan de gestion d'équipe et de projet

Yanis BOUHJOURA, Hugo DABADIE, Garance DUPONT-CIABRINI, Marek
ELMAYAN, Julien LALANNE

GL09 Groupe 2

26 janvier 2022

Table des matières

1	Description critique de l'organisation	3
1.1	Description de l'organisation	3
1.2	Analyse de l'organisation	3
1.2.1	Gestion des sprints	3
1.2.2	Utilisation de Jira	3
1.2.3	Daily meetings	4
1.2.4	Cohésion d'équipe	4
1.3	Pistes d'amélioration	4
2	Présentation de l'historique du projet	6
2.1	Ordre de conception et développement	6
2.1.1	Hello world	6
2.1.2	Langage sans objet	6
2.1.3	Langage objet	6
2.1.4	Extension TAB	7
2.2	Temps passé sur les différentes activités	7
2.2.1	Analyse	7
2.2.2	Conception	7
2.2.3	Codage	7
2.2.4	Validation	8
3	Synthèse et conclusion	8

1 Description critique de l'organisation

1.1 Description de l'organisation

Notre organisation d'équipe sur le projet s'est grandement inspirée de la méthode Agile Scrum. Nous avons utilisé le concept de sprints, en posant leur durée à environ une semaine. Nous avons dû adapter cela par rapport à la méthode Scrum qui préconise plutôt des sprints de deux à trois semaines, ce qui, compte tenu de la durée du projet, n'était pas envisageable.

De plus, nous avons utilisé un outil appelé Jira, conçu pour la méthode Agile et riche en fonctionnalités. Cela nous a permis d'organiser les sprints à l'aide, notamment, d'User Stories représentant les fonctionnalités du projet, elles-même découpées en Tasks, représentant les unités de travail restant à faire sur la fonctionnalité.

Nous avons également repris de la méthode Scrum les Daily Meetings, toujours en les adaptant. Ces réunions n'étaient pas à heure fixe mais devaient être quotidiennes. Elles avaient pour but de mettre en commun les difficultés et les avancées de chacun. L'objectif était de montrer à tous les membres du groupe l'avancée globale du projet car tout le monde ne travaillait pas sur la même partie et, de fait, n'était pas au courant de toutes les dernières avancées.

1.2 Analyse de l'organisation

1.2.1 Gestion des sprints

Notre gestion des sprints a été assez correcte. En effet, lors du premier sprint, pour lequel le but était de réaliser un *"Hello World"*, nous nous sommes vraiment consacrés à cela et nous n'avons pas commencé d'autres parties. De même, lors des deux autres sprints, *langage sans objet* et *langage objet avec extension*, nous ne sommes pas beaucoup sortis du périmètre du sprint.

1.2.2 Utilisation de Jira

L'utilisation de Jira s'est faite en trois étapes. Une première étape a été la prise en main des outils. La deuxième a été l'utilisation correcte de Jira et enfin, la troisième a été une phase où Jira n'était plus tellement utilisé.

Durant la phase de prise en main, qui a duré jusqu'à la première réunion de suivi à peu près, nous nous sommes approprié l'outil. En effet, au début, nos User Stories n'étaient pas axées fonctionnel mais plutôt technique, par exemple, *"faire l'étape A"*. Nous avons corrigé cela grâce, notamment, aux conseils reçus lors de la première réunion de suivi. Aussi, nous ne connaissions pas toutes les fonctionnalités, comme par exemple la possibilité de faire des Tasks dans des User Stories.

Lors de la deuxième phase, qui s'est terminée quelque temps avant la troisième réunion de suivi, nous avons utilisé Jira de manière assez classique et en respectant les principes Agiles tels que faire des User Stories axées fonctionnalités. Cela nous a permis de nous organiser dans le travail et de savoir à n'importe quel moment qui réalisait quelle tâche. On a ainsi évité que deux personnes travaillent sur la même chose.

Enfin, la dernière phase a été une période, survenue après la troisième réunion de suivi, dans laquelle nous avons utilisé de moins en moins Jira en raison du temps croissant investi pour faire des User Stories correctes. Nous nous sommes donc désintéressés de cet outil. Cependant, le temps gagné à ne pas créer ces User Stories a sûrement été perdu par la suite. En effet, il est arrivé que certaines personnes travaillent sur la même chose sans le savoir ou que l'on manque d'un cadre bien défini. Cependant, ces problèmes ont été palliés les derniers jours car nous nous sommes tous réunis afin de travailler ensemble.

1.2.3 Daily meetings

Les daily meetings ont été faits tous les jours jusqu'au milieu du projet Génie Logiciel et moins régulièrement après. Ils ont permis au début de maintenir un contact et de faire un point sur l'avancement du projet. Néanmoins, nous faisons ces réunions généralement à la fin d'une journée de travail. Par conséquent, le lendemain matin, nous ne nous souvenions pas toujours de ce qui avait été dit la veille.

Vers la fin du projet, surtout la dernière semaine, nous avons travaillé beaucoup plus ensemble en présentiel, que cela soit à l'Ensimag ou dans un de nos appartements. C'est pourquoi nous faisons moins de Daily meetings, nous nous voyions toute la journée. Toutefois, ces réunions auraient pu être bénéfiques, même si moins nécessaires lors du travail ensemble. Elles auraient permis notamment de cadrer davantage le travail effectué.

1.2.4 Cohésion d'équipe

La cohésion d'équipe était un point important de notre charte d'équipe. En effet, renforcer la cohésion d'équipe permet de renforcer l'envie de travailler et donc d'avancer sur le projet.

Nous avons effectué quelques actions de cohésion telles qu'une sortie au bar ou quelques repas partagés. Cela a permis à tous d'échanger en dehors du contexte de travail du projet et de renforcer les liens dans l'équipe. Les liens ainsi fortifiés ont permis une meilleure aide et une meilleure communication sur les tâches en cours.

1.3 Pistes d'amélioration

Lors de la rétrospective collective, nous avons défini quelques actions correctrices que nous pourrions réaliser si nous devons refaire le projet Génie Logiciel.

La première action à mettre en place serait de revoir l'utilisation de notre outil d'organisation. En effet, nous nous sommes rendu compte que Jira était un outil très puissant et polyvalent mais qu'il était peut être plus adapté pour les gros projets que pour les petits. Il aurait été bien qu'une personne se charge de mettre à jour régulièrement les User Stories ainsi que la backlog, c'est-à-dire presque avoir un rôle de Product Owner. Une autre solution aurait été d'utiliser un logiciel moins puissant mais plus simple d'utilisation tel que Trello.

La deuxième action votée a été sur le renforcement de la cohésion d'équipe en passant surtout par l'organisation de réunions type Daily Meeting avant de se mettre au travail. Il aurait donc fallu fixer des créneaux de réunions quotidiens le matin. Pour la cohésion d'équipe, il est ressorti que nous aurions aimé faire plus d'activités de groupe en dehors du cadre du projet. Nous souhaiterions donc fixer et organiser des sorties de cohésion hebdomadaires environ deux fois par semaine.

Enfin, la dernière action votée a été concernait la documentation. Il aurait été bénéfique de s'en occuper tout au long du projet. Les actions à mettre en place seraient donc une mise à jour régulière de la documentation par la personne s'étant chargée de la fonctionnalité en question et des revues de ces documentations avant chaque suivi lors de réunions dédiées afin que de s'assurer que ces dernières soient à jour.

Parmi les autres actions qui ressortaient mais qui n'ont pas été votées, on peut notamment citer le fait de ne pas laisser la réalisation des tests à une seule personne mais que celle-ci soit faite par tous les membres de l'équipe. On peut également citer qu'il aurait mieux fallu planifier les tâches à effectuer dans chaque sprint afin de mieux cadrer le travail.

2 Présentation de l'historique du projet

2.1 Ordre de conception et développement

2.1.1 Hello world

Nous avons commencé le développement par la réalisation de l'implémentation d'un programme affichant le message *"Hello World"*. Cette première étape a surtout été le moyen de comprendre le projet et ce qu'il fallait faire. Sur cette étape, la difficulté n'était pas réellement d'ordre technique mais plutôt d'identification du travail à réaliser. En effet, la quantité d'informations à apprendre et à comprendre était très importante.

Durant cette phase, tous les membres du groupe ont commencé par s'appropriier les spécifications demandées. Il n'y a pas eu réellement de répartition du travail, le but étant que chacun puisse trouver ce sur quoi il était le plus à l'aise.

2.1.2 Langage sans objet

L'objectif de cette étape était de réaliser la partie du langage Deca qui ne comportait pas d'objet. C'est-à-dire qu'à l'issue de cette étape il est possible de créer de réels programmes qui peuvent être assez complexes.

Nous avons donc commencé à nous spécialiser à partir de cette phase-là. En effet, chacun avait ses préférences et est donc devenu plus expert que les autres dans sa partie :

- Julien a essentiellement fait l'étape A, correspondant au lexer et au parseur, puis est venu aidé sur les étapes B et C ;
- Yanis a surtout commencé à réfléchir à l'extension et à comment nous pouvions concevoir celle-ci ;
- Marek a travaillé sur l'étape C, correspondant à la génération du code assembleur ;
- Garance a travaillé sur les tests en créant notamment des scripts permettant de lancer tous nos tests et d'avoir un retour exploitable ;
- Hugo a en grande partie travaillé sur l'étape B, correspondant à la vérification contextuelle lors de la compilation.

2.1.3 Langage objet

L'objectif de cette partie du projet était d'implémenter le langage objet et donc terminer l'implémentation du langage Deca complet. À l'issue de cette phase, il est possible d'utiliser tout ce qui est dans les spécifications du langage Deca.

La répartition des tâches est restée relativement similaire à la partie sans objet du fait que chacun avait des connaissances plus précises dans sa partie du projet.

2.1.4 Extension TAB

Le but de cette phase était d'étendre le langage Deca avec l'implémentation de tableaux et de matrices. L'objectif était aussi d'implémenter une bibliothèque de calcul matriciel permettant de réaliser différentes opérations sur les matrices.

Concernant la répartition des tâches sur l'extension, seul Yanis a commencé à travailler dessus avant les cinq derniers jours du projet. Hugo l'a ensuite rejoint en partie puis les autres membres ont commencé à travailler dessus que quelques jours avant la date de rendu, du fait de la volonté de terminer l'objet.

Cette répartition des tâches et le temps consacré à l'extension sont les raisons pour lesquelles il a été difficile d'implémenter cette partie. La plupart des fonctionnalités sont présentes mais quelques problèmes persistent. Il aurait sûrement été bénéfique d'y accorder plus de temps et de faire une analyse et une conception plus profonde sur cette partie.

2.2 Temps passé sur les différentes activités

2.2.1 Analyse

Nous avons essentiellement fait l'analyse au début du projet, durant la première semaine. Cette phase nous a pris à peu près une dizaine d'heures chacun. Nous avons également refait une phase d'analyse lors de l'extension mais qui aurait dû être plus poussée. Le temps consacré est d'environ cinq heures au total.

2.2.2 Conception

La conception est une partie de l'implémentation que nous avons un peu négligée. Néanmoins, nous avons essayé de passer un peu de temps sur cette étape essentielle pour tout projet d'une telle envergure. Le temps consacré est d'environ une quinzaine d'heures au total sans l'extension et cinq heures pour l'extension.

La conception étant déjà en partie faite sur le langage Deca, cela n'a pas forcément été préjudiciable mais, sur l'extension, nous nous sommes rendu compte que nous aurions dû passer plus de temps sur cette phase.

2.2.3 Codage

Le codage est la partie du projet qui nous a pris le plus de temps. En effet, on l'estime à plus de deux cent heures de travail au total. Nous nous sommes focalisé sur cette partie qui nous a semblé primordiale.

2.2.4 Validation

La validation est la deuxième partie du projet qui nous a le plus occupé. Garance a été à temps plein dessus. On estime donc à quatre-vingt heures de travail de sa part sur cette partie et une centaine d'heures au total le temps passé sur cette partie.

3 Synthèse et conclusion

Le projet de Génie Logiciel de l'Ensimag est l'un des plus, si ce n'est le plus, formateurs que nous ayons jamais fait dans notre parcours scolaire. En effet, la durée du projet, trois semaines à temps plein, et le nombre de membres de l'équipe, cinq, font de ce projet une très bonne introduction au travail dans le monde professionnel.

Nous avons découvert le travail en méthode Agile et sa mise en place avec également ses difficultés. Toutefois, ce que nous retenons sur la méthode Agile est notamment l'intérêt de celle-ci et de toutes ces cérémonies, artefacts ou autres. En effet, dès lors que nous n'appliquions plus un des principes de la méthode, nous rencontrions des difficultés. Au-delà d'avoir mis en pratique la méthode Agile, ce projet nous a fait comprendre l'intérêt de celle-ci.

D'un point de vue technique, ce projet nous a permis de consolider nos compétences sur les technologies utilisées mais surtout de nous apprendre la méthode de conception de logiciel avec les différentes phases comme l'analyse, la conception, le codage et la validation. Tout comme la gestion de projet, nous avons cerné l'intérêt de ces principes de développement et leur utilité au-delà de les avoir simplement appliqués.

Nous sommes donc convaincus que ce projet aura été très formateur sur plusieurs points concernant le génie logiciel et nous sera très utile pour notre future carrière.