



PHELMA GRENOBLE INP
ENSIMAG GRENOBLE INP

Bilan de la gestion d'équipe

DIJS Thomas FALGAYRAC Loic HO-SUN Jules
NOIRY Sylvain WANG Caroline
groupe GL 54

Phelma SEOC/ Ensimag ISI-MMIS
January 26, 2022

Table des matières

1	Introduction	1
2	Présentation de l'équipe	1
2.1	L'équipe : ses points forts	1
2.2	L'équipe : ses points faibles	1
3	Historique du projet	2
3.1	Gestion du temps et avancement du projet	2
3.2	Timeline du projet	2
3.2.1	Appropriation du sujet avec l'étape 1 "Hello world" . . .	2
3.2.2	Implémentation de l'étape 2 "sans objet"	2
3.2.3	Etape 3 "avec objet" et extension OPTIM	3
3.3	Difficultés rencontrées sur le projet	3
4	Enseignements	3
4.1	Enseignements communs	3
4.2	Enseignements personnels	3
4.2.1	Caroline WANG	3
4.2.2	Loïc FALGAYRAC	3
4.2.3	Thomas DIJS	4
4.2.4	Jules HO-SUN	4
4.2.5	Sylvain NOIRY	5

1 Introduction

Le projet Génie Logiciel est un projet proposé par l'Ensimag tous les ans. Ce projet se déroule sur 3 semaines et réunit des équipes de 5 élèves. Le but de ce projet est de créer un compilateur pour le langage Deca en suivant un cahier des charges et des contraintes spécifiques, mais il sert avant tout à apprendre la gestion d'une équipe et d'un projet. Le travail demandé est découpé en 3 étapes et chaque étape comporte 3 parties. D'une étape à une autre, nous devons reprendre les parties de l'étape précédente et les développer ; chaque étape constitue un livrable potentiel pour le client. En découpant le travail ainsi, nous utilisons une méthode Agile de développement. Le projet s'est relativement bien passé, nous avons fini le compilateur et réussi à implémenter l'extension OPTIM à ce dernier.

2 Présentation de l'équipe

2.1 L'équipe : ses points forts

La charte d'équipe écrite en début de projet a permis de mettre en place des conditions de travail dans lesquelles chaque personne du groupe se sentirait bien, un point d'honneur a été mis sur la communication. Grâce à cette charte, le projet s'est toujours passé dans la bonne ambiance, il n'y a eu aucun conflit interne au groupe ni de vrai désaccord. Le stress a su être utilisé à bon escient lorsqu'il se faisait ressentir, souvent avant les dates de rendu. La gestion des risques mise en place concernait surtout les potentielles absences dues à la crise sanitaire du Covid-19, par chance personne n'a été malade pendant le projet, nous n'avons donc jamais eu de retard conséquent. Nous avons quand même prévu des outils ne permettant de travailler en distanciel tels que Git et Discord, bien qu'un outil tel que Git soit quasiment indispensable pour un tel projet. Nous avons été contraints de travailler en distanciel via Discord le temps d'une journée pour cause d'impossibilité d'accès aux salles de l'Ensimag et cette journée en distanciel n'en a pas été moins productive pour autant.

Le fait que certaines personnes ne se connaissaient pas au début du projet n'a pas du tout été un problème, et a au contraire sûrement permis de se concentrer plus rapidement sur le travail à réaliser.

2.2 L'équipe : ses points faibles

Le majeur point faible pour notre équipe est sûrement la ponctualité. Nous n'avons pas pris ce problème en compte dans la matrice SWOT mais il s'est avéré que c'en était bien un.

Certains élèves étant élèves à Phelma, ils avaient cours les lundis 17 et 24 Janvier, le travail a donc dû être réparti en conséquence.

Le codage assembleur représente une partie majeure du travail dans le projet et seulement une personne du groupe était familière avec ce langage, il a donc été compliqué de se mettre à niveau dans le temps imparti pour pouvoir l'aider de manière efficace.

3 Historique du projet

3.1 Gestion du temps et avancement du projet

La mise en place du planning a été laborieuse, il a été difficile de dégager dès le départ toutes les tâches à réaliser jusqu'à la fin du projet. Nous avons donc dans un premier temps établi un planning sur les 10 premiers jours couvrant le projet jusqu'à la partie "sans objet" ; puis, au bout d'une semaine, sans avoir fini la partie "sans objet" nous avons eu le recul nécessaire sur les tâches à faire pour arriver jusqu'à la fin du projet. Chaque tâche était découpée sur un temps limité et à réaliser par un nombre de personnes spécifiques ; elles pouvaient s'étendre d'une demi-journée à plusieurs jours et dépassaient rarement une semaine. Le planning prévisionnel collait presque parfaitement au planning réel au détail près que pour les parties "sans objet" et "avec objet", la rédaction de la génération du code assembleur (partie C) était terminée avant la vérification de l'arbre (partie B). Les tests et le débogage pour la partie C n'ont donc pas pu être faits directement.

3.2 Timeline du projet

3.2.1 Appropriation du sujet avec l'étape 1 "Hello world"

(Du 03/01 au 06/01)

- Création de la charte d'équipe
- Lecture du sujet et compréhension des attentes sur le compilateur
- Répartition des rôles en fonction des affinités de chacun avec les différentes tâches à accomplir
- Début du développement du Lexer et du Parser en fonctions des règles lexicographiques et syntaxiques attendues
- Compréhension de l'analyse contextuelle et de la génération de code assembleur (parties B et C)
- Réflexion sur le choix de l'extension (l'équipe a choisi OPTIM)

3.2.2 Implémentation de l'étape 2 "sans objet"

(Du 07/01 au 13/01)

- Rédaction de la fin du Lexer (quasi-identique pour les étapes 2 et 3) et début des tests du Lexer
- Développement du Parser et début des tests du Parser "sans-objet"
- Développement de l'analyse contextuelle
- Définition et implémentation de l'architecture logicielle et des composants de base pour la génération de code
- Développement de la génération de code assembleur pour deca sans objet
- Réflexion sur la génération de code pour deca avec objet
- Réflexion sur la manière d'optimiser le code pour l'extension OPTIM, avec une implémentation minimaliste d'un graphe de contrôle de flot

3.2.3 Etape 3 "avec objet" et extension OPTIM

(Du 13/01 au 24/01)

- Fin du développement du Parser et finalisation des tests du Parser
- Fin du développement de l'analyse contextuelle et de la gestion des erreurs
- Rédaction de tests de l'analyse contextuelle
- Fin du développement de la génération de code assembleur pour deca avec objet, en réutilisant l'architecture déjà existante faite la semaine précédente
- Rédaction de tests de la génération de code
- Implémentation de l'extension OPTIM, avec la mise en forme SSA, la propagation de constantes, et la suppression de code mort

3.3 Difficultés rencontrées sur le projet

Le projet étant très conséquent, les horaires que nous nous étions fixés sur la charte d'équipe n'étaient clairement pas suffisants (9h-12h / 13h30-17h30). Nous avons donc dû travailler au-delà de ces horaires pour pouvoir rendre notre travail à temps sur chacune des dates de rendu imposées, ce qui impliquait parfois de travailler tard le soir. C'est également ce qui explique le manque de ponctualité le matin.

4 Enseignements

4.1 Enseignements communs

Ce projet nous a apporté beaucoup de connaissances : il pousse la compréhension du compilateur au plus bas niveau de programmation qu'est le code assembleur. C'est également notre premier projet à temps plein sur une durée aussi "longue". Au-delà du fait qu'on nous demande de mettre en place de la gestion de projet, nous nous sommes aperçu que cette gestion est nécessaire. Cette mise en place de gestion de projet est également une première pour certains d'entre nous. Nous avons appris à travailler en équipe, à utiliser une méthode de développement dite "Agile", à communiquer sur nos capacités à devoir livrer du code qui fonctionne aux autres membres de l'équipe et à répondre à leurs questions en cas d'incompréhension ou de bug sur notre partie. Les quelques manques de communication ayant eu lieu lors des dates butoir nous ont rappelé à quel point cette notion est importante dans un travail d'équipe.

4.2 Enseignements personnels

4.2.1 Caroline WANG

4.2.2 Loïc FALGAYRAC

Avant de commencer le projet, j'en avais déjà beaucoup entendu parler, et le polycopié de 200 pages était à la fois intimidant et mystérieux. Dès le début du projet, nous avons rapidement su faire connaissance et instaurer une ambiance agréable dans

l'équipe. J'ai eu beaucoup de mal à comprendre l'implémentation du projet lorsque j'ai débuté avec la partie B, ce qui a engendré une peur de ne rien comprendre. Ensuite, j'ai attaqué la partie C avec Sylvain et ai pu comprendre en détail son architecture et participer à son implémentation. C'est à ce moment que je me suis rendu compte que l'implémentation était moins difficile qu'imaginée.

C'est le premier projet aussi important sur lequel je travaille, et je pense que cela a rendu l'expérience encore plus intéressante et importante vis-à-vis de ce que je pourrai faire plus tard, et cela sort assez des *rails* de l'apprentissage traditionnel, d'où sa valeur ajoutée. Enfin, on se rend compte que ce que nous reléguons souvent au second plan (planning, gestion de projet ...) pour se concentrer sur la partie logiciel prend finalement une place importante et permet d'éviter de travailler inutilement ou de s'égarer.

4.2.3 Thomas DIJS

J'ai commencé ce projet avec un peu d'appréhension car je ne savais pas du tout à quoi m'attendre, je n'avais donc pas conscience du travail qui était attendu. Ne connaissant pas les étudiants de l'Ensimag, je ne connaissais pas du tout leur méthode de travail ni les compétences qu'ils avaient. En venant de Phelma, une école bien moins spécialisée en programmation, je pensais que j'allais commencer le projet en ayant du retard sur la compréhension de ce qui était demandé par rapport à mes camarades de l'Ensimag. J'ai donc choisi de me pencher sur la partie A qui demande à rédiger le code du Lexer et du Parser, une partie certes longue, mais relativement simple quand on comprend les attentes du travail demandé. J'avais déjà eu l'occasion auparavant de travailler sur des projets de groupes (4 et 8 personnes) mais je n'avais jamais expérimenté de réelle gestion de projet, c'est selon moi l'enseignement majeur que j'en tire de ce projet. Je n'avais, par conséquent, jamais travaillé en suivant une méthode Agile, je l'ai trouvée très efficace puisqu'elle permet d'avoir un livrable très tôt dans le projet, et de développer ce livrable au long du temps imparti.

N'étant pas trop sujet au stress, j'ai très bien vécu ces 3 semaines de projet, le fait de travailler dans un école où je ne connais personne m'a permis de me concentrer sur ma tâche et donc d'être efficace sur mes heures de travail. La partie de laquelle j'étais chargée étant la "racine" du projet, mes camarades en avaient impérativement besoin pour pouvoir avancer, cela m'a donc poussé à travailler plusieurs fois jusqu'à 00h00 (sans aucune contrainte de la part de mes camarades).

4.2.4 Jules HO-SUN

N'étant pas de nature très anxieuse, je n'avais aucune appréhension du projet. J'ai aussi eu la chance d'avoir des liens antérieurs au projet avec tous les membres de l'équipe ; communiquer n'a donc pas été pour moi une difficulté, y compris lorsque nous venions de commencer à travailler en groupe. Grâce à la très bonne ambiance dans l'équipe, je n'ai pas senti ces trois semaines passer : ce fut une très bonne expérience, autant sur le plan de l'enseignement que sur le plan humain.

Ces quelques semaines de travail intensif m'ont permis d'apprendre de nombreuses choses, mais le plus intéressant a sûrement été le fonctionnement que nous avons dû mettre en place pendant le projet. En effet, je ne pense pas dans le futur travailler sur des compilateurs ; néanmoins, je pense que la division du travail en différents sprints est une méthode qu'il est très utile de connaître. Diviser ce qui doit être fait en plusieurs parties est sûrement ce qui est le plus efficace pour réaliser un travail efficacement

et dans les temps, autant d'ailleurs dans un cadre professionnel que dans un cadre personnel. Si parfois une communication claire entre tous les membres de l'équipe n'a pas été très facile, ces moments de difficulté m'ont permis de comprendre à quel point la communication est importante durant un projet avec autant de membres dans l'équipe : la moindre défaillance dans le système de réunions et de communication que nous avons mis en place nous a sûrement fait perdre un précieux temps.

Ma seule déception, qui n'en est pas est vraiment une, est que nous n'avons pas partagé beaucoup de moments autres que le travail ; les quelques moments durant lesquels nous avons discuté d'autres sujets ou partagé des gâteaux étaient très appréciables. Il aurait sûrement été très sympathique d'en avoir plus.

4.2.5 Sylvain NOIRY

Avant de commencer le projet GL, j'avais quelques attentes sur ce que cela pouvait m'apprendre. C'était l'un des premiers projets hors du cadre académique que j'ai eu l'occasion de faire, je voulais notamment pouvoir me familiariser avec les problématiques de gestion d'équipe et la méthode agile, vantée par tant de personnes. Je n'ai pas été déçu sur ces aspects, puisque j'ai particulièrement apprécié les cours que nous avons eu sur la gestion de projet, et l'application que nous en avons faite durant le projet. Certes notre organisation n'était pas parfaite, nous aurions par exemple pu mieux délimiter les sprints et plus utiliser le tableau SCRUM, mais je suis plutôt content de cette approche. Je trouve le découpage en sprints très sécurisant sur notre capacité à rendre un produit minimum, et permet de se donner un cap clair sur un horizon proche. Néanmoins les dépendances entre parties a été difficile à gérer, puisqu'il fallait attendre les étapes en amont afin de tester la suite. Ainsi, alors que j'ai spécialisé dans la génération de code, j'avais souvent un début de sprint peu chargé durant l'implémentation et l'attente des parties précédentes, mais la fin de sprint était au contraire très dense pour pouvoir debug et tester avant les rendus. Je trouve que nous n'avons pas assez pu testé la génération de code en particulier, ce qui est un peu frustrant.

Notre organisation d'équipe nous a conduit à avoir un expert pour chacune des principales parties. Même si nous voulions avoir au moins deux personnes capables à tout moment de prendre en charge une partie, nous n'avons pas vraiment réussi à le respecter. Ainsi alors que j'étais notamment en charge de l'extension, j'ai pu ressentir la pression et la responsabilité de n'avoir aucune aide en cas de problème et de devoir réussir seul à produire quelque chose de fonctionnel. Les deadlines étant assez serrées, j'ai aussi eu parfois peur que les parties en amont de la chaîne de compilation prennent du retard et paralysent le projet. Nous aurions alors dû tous se mettre à debug ces point en particulier, que je ne maîtrisais pas très bien.