



UGA – Grenoble INP

Projet Génie Logiciel

Bilan de Gestion d'équipe et de projet

Grenoble, 25 janvier 2022.

Table des matières

1	Retour sur le planning	3
1.1	Planning initial	3
1.2	Planning final	3
2	Description des phases de l'implémentation	4
2.1	Première phase : phase d'analyse	4
2.2	Deuxième phase : développement du compilateur sans objet	5
2.3	Troisième phase : déploiement du compilateur sans objet et réalisation de l'extension	5
2.4	Quatrième phase : phase finale	6
3	Retour sur l'organisation du groupe	6
3.1	Organisation initiale	6
3.2	Organisation finale	7
4	Problèmes rencontrés	7
4.1	Problèmes d'organisation	7
4.2	Problèmes de communication	8
4.3	Problèmes de répartition des tâches	8
5	Conclusion	8

1 Retour sur le planning

Le planning initial présenté a été amené à évoluer en fonction des défis et obstacles que nous avons rencontrés lors de la conception du projet. Voici les deux versions (initiale et finale) du planning : il y a eu des modifications intermédiaires.

1.1 Planning initial

Voici le planning initial, tel que présenté au tout premier point de suivi du 05 janvier 2022 :

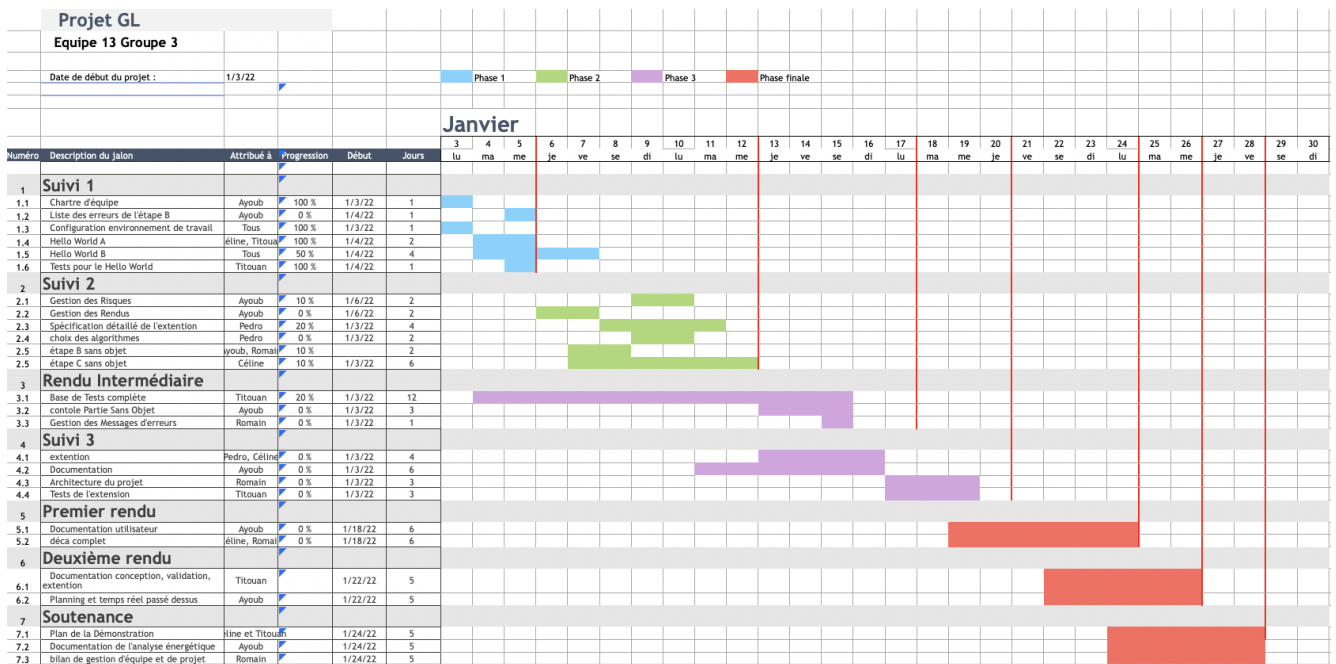


FIGURE 1 – Planning initialement prévu

Nous pouvons voir que ce planning était un peu optimiste, il prévoyait la fin effective de la programmation de tout le logiciel avant le 19 janvier 2022. Cela s'est vite avéré quelque peu serré comme deadline.

Ce planning a été modifié avant chaque point de suivi avec les encadrants du projet. En effet, entre chaque rendu, des difficultés à la fois techniques et organisationnelles s'ajoutaient et nous obligeaient à revoir notre planning.

1.2 Planning final

La Figure 2 dans la page ci-après montre l'état de notre planning final, présenté lors du point de suivi du 19 janvier 2022. Il y a de nombreux changements à remarquer sur ce planning.

Premièrement, notons que dans ce planning final l'idée de livrer un logiciel complet a été abandonnée. En effet, l'extension que nous avons prévu d'achever le 16 janvier dans le planning initial, a été largement repoussée dans le planning final, jusqu'au 24 janvier, date de livraison du logiciel. L'extension ne sera donc pas développée au complet. Notre compilateur Jasmin (extension Bytecode) ne pourra pas compiler des programmes avec objets, mais pourra réaliser pratiquement toute la partie sans objet (cf documentation de l'extension).

De plus, la majorité des parties a été repoussée de quelques jours, du fait que la partie C sans objet ait été repoussée d'un jour.

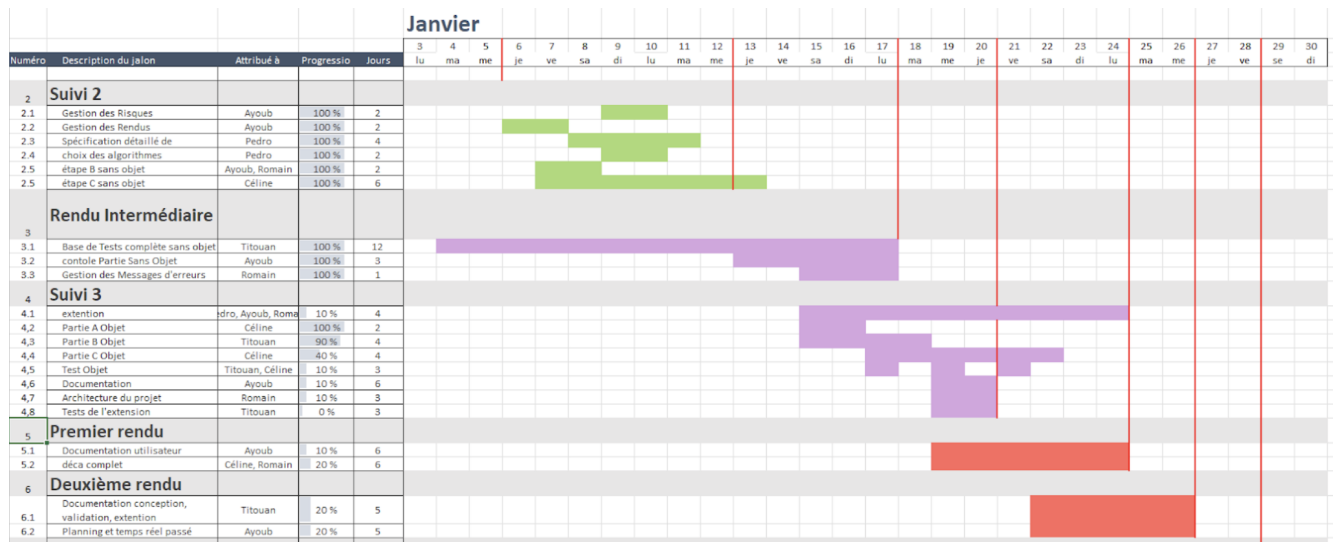


FIGURE 2 – Planning final

2 Description des phases de l'implémentation

Dans le cadre d'une gestion de projet suivant la méthode Agile, nous avons pris le soin de découper le travail à réaliser en différentes phases complémentaires dans le but de faciliter le développement en équipe et surtout de déceler les problèmes aussi petits soient-ils pour mieux les appréhender.

Ainsi, le projet a été découpé en quatre phases initialement :

2.1 Première phase : phase d'analyse

La première phase de développement correspond au premier contact avec le projet, la mise en place d'un environnement de travail, la mise en place d'un climat de travail au sein de l'équipe et la répartition des tâches sur les semaines à venir.

Ainsi, lors de cette phase, les premiers outils de communication de l'équipe ont vu le jour : il s'agit de notre serveur Discord et d'un groupe sur Messenger. Cela nous a permis de travailler à distance, tout en nous rencontrant ponctuellement dans les locaux de l'Ensimag lorsque le besoin se présentait. Un dépôt Gitlab hébergé sur les serveurs de l'école a aussi été mis en place lors de cette phase, avec l'usage de plusieurs branches pour permettre à chacun de coder en parallèle.

Du côté du code, cette première phase marque le départ de la phase de développement. Bien qu'une bonne partie de celle-ci soit en réalité une phase d'analyse, une partie du code a aussi été réalisée. Ainsi, en plus de la mise en place de la charte d'équipe et du planning prévisionnel, la partie A du compilateur sans objet a été développée, ainsi que la liste des erreurs pour l'étape B du même compilateur. Enfin, la phase a été clôturée par la mise en place des premiers tests et la compilation des tout premiers programmes : il s'agit tout naturellement du bien célèbre programme "HelloWorld".

Toutefois, à l'issue de cette phase, nous avons remarqué que la partie B prendrait plus de temps que prévu, sachant que la partie C est quant à elle à grandement dépendante de celle-ci. Cela nous a conduit à notre première prolongation du planning prévisionnel.

2.2 Deuxième phase : développement du compilateur sans objet

Lors de cette deuxième phase, le travail de l'équipe s'est porté sur la partie sans objet du compilateur. Elle marque la fin de toute la partie sans objet. Par ailleurs, cette phase a été marquée aussi par le début du travail sur l'extension choisie (génération de Bytecode).

Ainsi, lors de cette phase, les étapes B et C du compilateur sans objet ont été réalisées, permettant ainsi d'avoir en mains toutes les pièces du puzzle du langage sans objet. L'étape B a donc pris plus de temps que prévu comme spécifié auparavant, ce qui décala systématiquement la réalisation de l'étape C qui en dépendait fortement. Mais à l'issue de cette phase, les parties ont été correctement mises en place, ce qui justifia malgré tout le petit retard engendré.

En parallèle, les premiers travaux sur l'extension ont été réalisés : il s'agit en premier lieu d'une phase de recherche sur l'extension choisie. Ainsi, il a été décidé - et par recommandation de l'encadrant - qu'au lieu de générer du Bytecode, nous allions plutôt générer du code Jasmin. Le choix de la structure et l'architecture globale des programmes de génération de Jasmin ont été réalisés lors de cette phase, ainsi que le choix des algorithmes.

2.3 Troisième phase : déploiement du compilateur sans objet et réalisation de l'extension

Lors de cette phase, plusieurs parties du projet ont été réalisées et ont donné lieu aux premiers rendus conséquents du logiciel. Cette phase a donc été riche en développement en tous genres, en tests et en déploiements. Elle a aussi permis de réaliser à quel point notre logiciel serait avancé à la fin du projet, et nous a donc permis de nous fixer les limites théoriques de notre compilateur final.

D'une part, cette phase a connu un développement massif de tests, pour tous les programmes que notre compilateur sans objet devrait être en mesure de compiler en fichiers assembleurs. Ces tests ont donc été conçus de telle sorte que tous les types développés et toutes les fonctions de génération de code soient appelés pour tester notre compilateur en profondeur.

D'autre part, la finalisation du compilateur sans objet a eu lieu, en rassemblant toutes les pièces réalisées lors de la deuxième phase. Une journée de code a été spécialement dédiée dans le planning pour la gestion des messages d'erreurs. Suite à cela, le compilateur sans objet est alors complètement fonctionnel. C'est Céline qui nous annonça cela sur le groupe Messenger, le 13 janvier 2022. Cela a marqué un véritable tournant dans le projet, car pour la première fois depuis le début du projet, nous avons un véritable compilateur fonctionnel.

De plus, les travaux sur l'extension avançaient de leur côté, de manière assez indépendante du projet, mais s'inscrivant toutefois dans la continuité de l'étape C du compilateur. C'est pour cela que l'implémentation de l'extension n'a pas eu lieu avant la fin de ladite étape. Les travaux de l'extension ont donc eu lieu essentiellement en parallèle avec ceux du compilateur avec objet. Cela nous a permis entre autres de nous fixer une limite au niveau du projet. Il nous est alors paru improbable de mettre en place l'extension pour des programmes avec objets.

Enfin, cette phase a connu le début des différentes documentations du projet, qui seront livrées en fin de projet (dont le document ici-présent).

2.4 Quatrième phase : phase finale

Cette phase marque officiellement la fin du projet, que ce soit en terme de programmation ou de rendus en tous genres. Elle représente le point final des quatre semaines de travail et comprend en particulier la livraison du rendu final au Client. C'est aussi la phase durant laquelle le doute a le plus plané dans l'équipe, ainsi que le stress, l'incertitude et l'anxiété.

Lors de cette phase, il y a eu le développement du compilateur avec objet. Le compilateur sans objet était alors totalement fonctionnel. Il s'agissait de rajouter la possibilité de reconnaître et de compiler les objets nouvellement créés. De plus, les différentes documentations ont été réalisées et finalisées lors de cette phase.

Enfin, cette phase sera marquée par la fin effective du projet, et la réalisation de la soutenance devant les encadrants du groupe.

3 Retour sur l'organisation du groupe

Une organisation quelque peu approximative a été prévue en début de projet. Toutefois celle-ci s'est vite avérée inefficace. En effet, nous avons prévu en tout début de projet une organisation pointilleuse sur les différentes semaines de travail. Celle-ci n'a pas pu suivre le rythme Agile de notre philosophie d'implémentation.

3.1 Organisation initiale

Comme précisé ci-avant, l'organisation du développement était prévue au jour et à la personne près. Dans le planning initialement prévu (cf Figure 1), on peut constater que la disposition des différentes tâches a été prévue sur la longueur du projet. Ainsi, la répartition initiale entre les membres du groupe était la suivante :

- **Céline** devait initialement s'occuper des parties A et C du compilateur sans objet, ainsi que le passage à la partie avec objets, et d'une partie de l'extension.
- **Titouan** devait initialement s'occuper de différents tests, le HelloWorld A en particulier (qui a été le tout premier test compilé) et de tous les autres tests en général. Il devait ainsi travailler entre autres sur la mise en place de tests de l'extension, du compilateur Deca avec objet et sur la rédaction de quelques documentations.
- **Romain** devait initialement réaliser l'architecture du projet et veiller au respect de celle-ci, en plus de réaliser l'étape B de la partie du compilateur sans objet. Il devait aussi gérer les messages d'erreurs et les options de compilation, ainsi que la réalisation du compilateur Deca complet.
- **Ayoub** devait initialement travailler avec Romain sur l'étape B du compilateur sans objet. Il devait aussi réaliser la partie contrôle du compilateur sans objet. De plus, il est en charge de rédiger les différentes documentations, ainsi que la gestion des risques et la gestion des différents rendus.
- **Pedro** devait initialement travailler exclusivement sur l'extension, en réalisant en particulier toute sa documentation, en choisissant les algorithmes adéquats et en mettant en place l'extension sous la supervision de Céline.

3.2 Organisation finale

Finalement, et comme prévu, l'organisation a été modifiée. Cependant, quelques événements au sein de l'équipe ont mené à la révision de la distribution des tâches.

Ainsi, l'organisation finale a été la suivante :

- **Céline** a réalisé ses parties comme prévu, elle est aussi venue en renfort au niveau de la partie B sans objet. Toutefois, elle a été congédiée de son rôle initialement prévu au niveau de l'extension. Elle a réalisé une bonne partie du code, et sans elle l'équipe n'aurait jamais réussi à finaliser le compilateur.
- **Titouan** a aussi réalisé ses tâches initialement prévues en réalisant une couverture complète de tests pour l'ensemble des fonctionnalités. Il est aussi venu en renfort au niveau de l'étape B du compilateur sans objet. Il a aussi réalisé, avec Céline, le compilateur avec objet. Tout comme Céline, s'il n'était pas aussi investi, le compilateur n'aurait jamais vu le jour.
- **Romain** a réalisé son travail en tant que Git Master du groupe : il a réalisé toutes les fusions de branches et résolu tous les problèmes qui en découlaient. Il a toutefois été déchargé de l'étape B qui lui était initialement destinée avec Ayoub, pour se concentrer sur l'implémentation de l'extension.
- **Ayoub** a réalisé en partie le travail qui lui était destiné par le premier découpage des rôles. Il a été déchargé de l'étape B du compilateur sans objet tout comme Romain, pour se focaliser sur l'extension qui était assez conséquente au niveau de l'implémentation. Il a toutefois réalisé la rédaction de quelques documentations avec Pedro.
- **Pedro** était initialement destiné à réaliser la partie propre à l'extension au niveau du projet. Il a réalisé une excellente analyse de celle-ci, et a proposé des pistes très prometteuses au niveau de l'implémentation, en proposant de passer par Jasmin pour la génération des fichiers .class. Il a ainsi participé à l'implémentation de l'extension, et la rédaction de quelques documentations.

4 Problèmes rencontrés

Comme constaté dans la section précédente, l'organisation de l'équipe a souffert de beaucoup de changements imposés par différents facteurs internes et externes à l'équipe.

4.1 Problèmes d'organisation

Le problème capital au niveau de l'organisation initiale du projet, c'est qu'elle était découpée de façon bien trop grossière, et attribuée de façon trop arbitraire. En effet, l'ensemble du projet s'inscrivait dans une continuité, en commençant par l'étape A du compilateur sans objet et en finissant par l'étape C du compilateur avec objet. Par exemple, il était difficile à Romain et Ayoub de se pencher sur l'étape B sans avoir traité ni bien compris l'étape A. Cela a conduit à une première prolongation de la durée prévue pour cette partie en particulier.

Ainsi, le découpage initial de notre projet n'était pas du tout adapté, et il nous a fallu une bonne semaine avant de nous rendre compte que la meilleure solution était de programmer une seule et même partie tous ensemble et faire du *Pair Programming* au lieu de coder chacun de son côté et de se passer le témoin entre nous aux différentes parties du projet.

4.2 Problèmes de communication

La communication a été la bête noire de l'équipe durant tout le projet, depuis le tout début. Nous avons remarqué très tôt qu'il était difficile de réaliser des échanges très réguliers, pour une raison très simple : Titouan et Céline préféraient travailler dans les locaux de l'école pendant la journée, tandis que Romain, Ayoub et Pedro travaillaient à distance et majoritairement la nuit. Pour nos réunions de suivis prévues tous les jours, les horaires étaient alors très aléatoires et ne suivaient jamais un rythme régulier. Cela a eu comme fâcheuse conséquence de supprimer quelques réunions de suivis, ou de les réaliser en effectif réduit.

4.3 Problèmes de répartition des tâches

Dans la continuité de ce qui a été spécifié dans la sous-section 4.1, il y a eu un problème de répartition des tâches, en affectant des parties du projet aux personnes les moins qualifiées pour les réaliser. Ainsi, le projet avançait à des rythmes très hétérogènes et cela a influencé sur le climat global de l'équipe et sur les rendus prévus qui ont en partie été repoussés.

5 Conclusion

Le projet Génie Logiciel a permis à chacun d'entre nous d'apprendre quelque chose de nouveau, que ce soit au niveau technique ou au niveau du travail en équipe. En effet, nous avons tous réalisé pas mal de projets en groupe, ne serait-ce que depuis notre arrivée à l'Ensimag. Mais le projet Génie Logiciel est quand même assez différent, ne serait-ce qu'au niveau des défis, de la timeline et des apports technologiques qui en découlent.

Ainsi, malgré les différents problèmes qui ont surgi durant notre développement, nous avons tous appris de cette expérience, et chacun de nous a pu gagner en maîtrise technique et surtout en expérience de travail en groupe.