

Projet GL 2021:

Bilan de gestion d'équipe

*Équipe n°31 : Antoine Briançon, Axel Glorvigen,
Thibault Launay, Maxime Martin et Pierre Pocreau*

Sommaire:

I.	Introduction	p.1
II.	Organisation temporelle	p.1
III.	Répartition des tâches	p.3
IV.	Analyse des choix de gestion	p.4
V.	Conclusion	p.5
VI.	Annexe : la Charte d'équipe	p.6

I. Introduction

Ce document est une analyse de l'organisation que nous avons mise en place dans le cadre de la gestion d'équipe à cinq membres pour le Projet GL 2021. Nous allons ainsi présenter les méthodes d'organisation choisies, détailler quelques documents relatifs à cette organisation et effectuer une rétrospective générale sur les choix effectués en termes de gestion et d'organisation. Nous reviendrons ensuite sur les enseignements et expériences tirés de ce projet, à travers une analyse critique des choix réalisés.

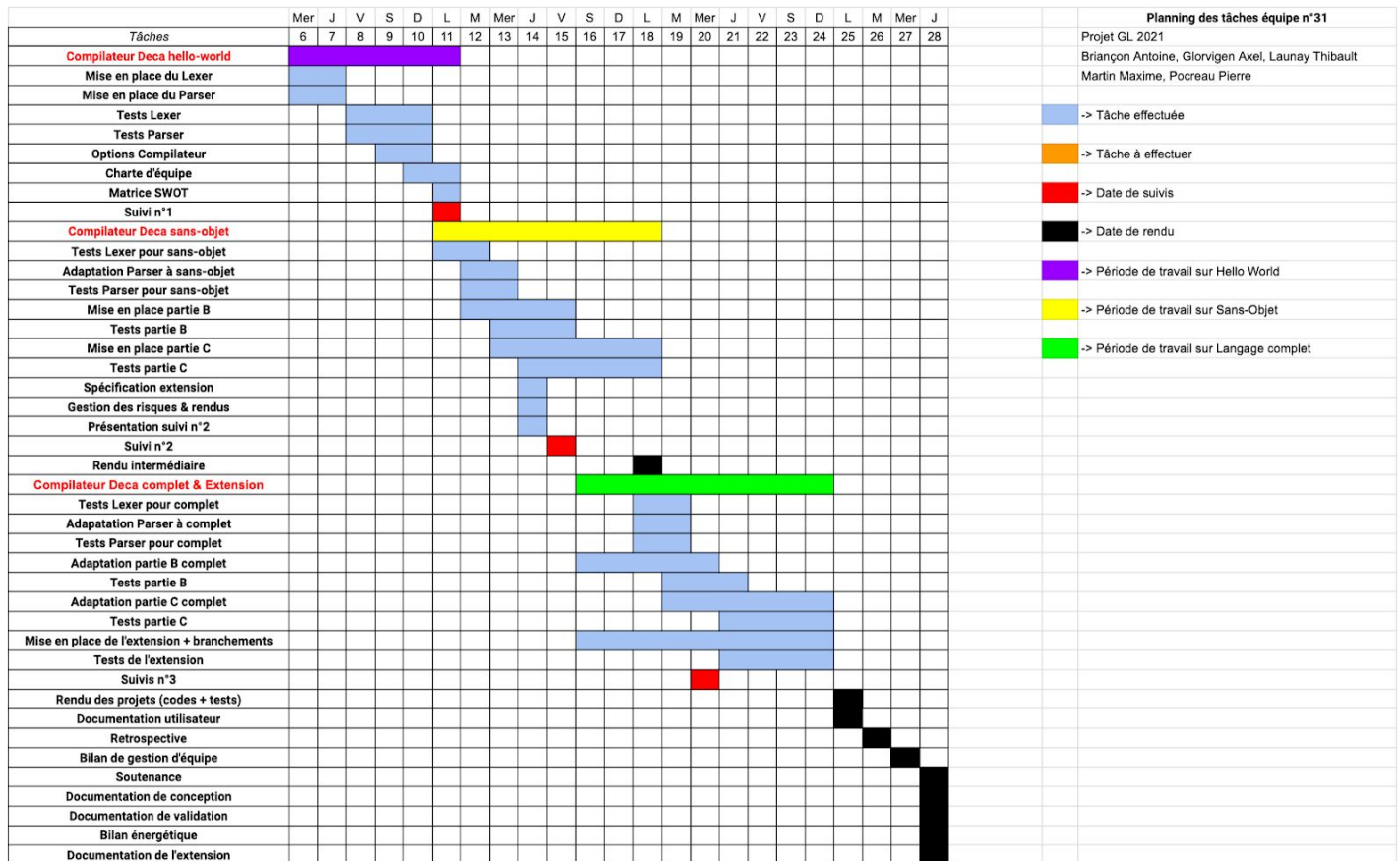
II. Organisation temporelle

Dans un premier temps, nous avons pris connaissance du projet, des tâches à effectuer, de l'environnement de développement et des dates imposées pour les rendus et soutenances orales. Cette première partie du travail nous a permis d'estimer le temps à passer sur certaines tâches ou parties du projet, afin d'avoir une vue à long terme sur l'avancement de nos travaux. Ce fil conducteur nous a permis de suivre l'évolution du projet de jour en jour, et d'ajuster le temps passé sur certaines tâches dans le but de présenter un produit complet lors des dates butoires.

Lors de la première semaine de travaux, nous n'avons pas mis en place de gestion particulière de l'équipe. En effet, nous continuions de chercher des méthodes d'organisation efficaces pour avoir une répartition optimale des tâches afin que chacun puisse avancer à son rythme, tout en découvrant de nouveaux travaux au fur et à mesure de la lecture des spécifications du projet. Ceci a aussi permis à chacun des membres de l'équipe de comprendre le sujet dans les grandes lignes et de réfléchir aux tâches qu'il souhaiterait effectuer. Nous avons conçu notre Charte d'Équipe (en annexe) pendant cette période, afin de nous accorder sur les règles et valeurs communes pour ce projet, limitant ainsi les risques de conflits liés à des avis divergeant sur des sujets relatifs au projet.

C'est lors de la deuxième semaine de travaux, après le premier entretien avec les professeurs, que nous avons décidé de suivre une organisation se rapprochant des méthodes agiles, avec la création d'un diagramme de Gantt simplifié et une répartition des tâches avec les outils d'organisation de la plateforme Gitlab. De plus, en termes de communication, nous avons décidé d'organiser une réunion quotidienne, sur la plateforme Discord, afin que tout le monde puisse suivre l'avancement du projet et partager ses problèmes de développement ou de compréhension. De même, nous nous sommes accordés pour utiliser l'application Messenger dans le cadre d'échange de messages et d'annonces.

Nous présentons ainsi le diagramme de Gantt mis en place pour la planification des tâches identifiées dans les spécifications du projet :



Ce diagramme, avec une précision au jour près, ne présente pas la répartition des tâches entre les membres de l'équipe, car cette répartition a été faite, comme expliquées précédemment, à l'aide des outils disponibles sur Gitlab. Nous pouvons cependant expliquer quelques points de ce diagramme, qui a été mis à jour quotidiennement pour garder un suivi régulier de l'avancement du projet.

Nous avons répartis nos trois semaines de travaux selon les propositions de la spécification, à savoir un découpage en trois langages de façon incrémentale : Hello World, Sans-Objet et Complet. Chacun de ses trois langages est aussi réparti en trois étapes majeures:

- Partie A : gestion de l'analyse lexicale et syntaxique
- Partie B : vérification et décoration de l'arbre du programme
- Partie C : génération du code assembleur

Chacune de ces étapes comprend une partie de compréhension, une partie de développement, une partie de tests et une partie de débogage. La répartition de ces étapes est visible sur le diagramme de Gantt, avec le temps en jours passés sur chacune d'elles. Nous avons aussi fait en sorte que les dates critiques, telles que les suivis et les rendus, soient visibles au premier coup d'œil, afin d'avoir rapidement

une estimation du temps restant avant ces échéances. Ce diagramme de Gantt est donc un outil indispensable pour l'organisation générale de l'équipe, afin de ne pas perdre de vue les objectifs et échéances du projet, mais il est cependant insuffisant pour rentrer dans le découpage détaillé de chaque tâche et pour évaluer rapidement les attributs des tâches aux différents membres de l'équipe.

La plateforme Gitlab héberge un système efficace et partagé permettant de faire un découpage précis des tâches en "issues", qui sont ensuite réparties sur différents "boards", où chaque "board" correspond à un langage (Hello World, Sans-Objet ou Complet). Chacun peut ensuite s'attribuer des "issues" et préciser des informations liées à cette "issue" : date de rendu, avancement, temps estimé, etc ... Une "issue" est fermée lorsque la tâche associée est complète, et que les tests ont bien été réalisés. Un board est fermé lorsque toutes les "issues" qui le composent sont fermées.

III. Répartition des tâches

Nous avons ainsi essayé de répartir les tâches afin que chacun puisse traiter plusieurs parties de projets, mais nous nous sommes finalement rendu compte que chaque partie nécessitait des connaissances bien précises acquises au fur et à mesure des travaux. Nous avons décidé d'attribuer une partie à chacun, afin de ne pas perdre de temps à réassimiler les bases à chaque fois que l'on revenait sur une partie. Nous avons donc eu le découpage suivant :

- Partie A: Pierre, Antoine Maxime
- Partie B: Axel et Pierre
- Partie C: Pierre, Thibault et Maxime
- Extension: Thibault, Axel et Antoine
- Tests: Antoine
- Documents de gestion d'équipe: Maxime

Nous avons, pour les partie B et C (la partie A n'étant pas très longue), décidé de mettre en place une implémentation en parallèle. Cela n'a pas posé de réels problèmes car nous pouvions tester le contexte indépendamment, cependant les tests de la partie C devaient être décalés en attendant la fin de la partie B pour fonctionner.

Nous nous sommes très vite rendus compte de l'importance des tests sur un projet d'une telle ampleur. Nous avons ainsi passé beaucoup de temps avec les

tests et le débogage pour prendre en compte toutes les spécificités du cahier des charges. On peut ainsi estimer que nous avons réparti notre de la manière suivante :

- 15% analyse
- 10% conception (une grande partie du squelette était fournie)
- 25% codage
- 35% validation et débogage
- 15% documentation

Sur le plan humain, nous n'avons pas eu de difficultés majeures en rapport avec le travail d'équipe. Tout le monde était présent régulièrement sur les différentes plateformes de communication (Discord, Messenger et Zoom). Notre équipe était soudée, et nous avons profité de l'appartement de Pierre et Axel, qui sont en collocation, pour pouvoir nous rencontrer physiquement malgré le contexte sanitaire.

Cette bonne dynamique de travail a permis d'aller de l'avant, de garder la motivation tout au long du projet malgré la contrainte du distanciel imposée cette année. Personne n'a décroché ou n'a pas fait une tâche qui lui était attribuée. Ceci a imposé de faire confiance aux membres de l'équipe afin que chacun fasse le travail demandé, car nous n'avions pas de chef d'équipe, essayant d'appliquer les règles des méthodes agiles.

IV. Analyse des choix de gestion

Nous sommes fiers de la réalisation que nous avons faite pour ce Projet GL 2021. Notre compilateur n'est pas parfait, et nous savons exactement ce qu'il reste à travailler pour l'améliorer. Cependant, nous pensons que nous avons bien su gérer le travail en équipe, à travers nos méthodes de gestion, alors que nous avons peu d'expérience dans ce domaine. Nous voudrions néanmoins revenir sur certains aspects gestionnels que nous aurions certainement pu améliorer, constat possible avec une prise de recul par rapport à la globalité du projet.

En effet, nous pensons que nous aurions certainement été plus efficaces si nous avions travaillé sur une organisation claire dès le début du projet, avec une analyse plus rapide du sujet et des tâches à réaliser. De même, travailler sur un diagramme de Gantt dès le début du projet nous aurait aidé à directement avoir une vue à long terme de ce que nous voulions réaliser, et non une vision arrêtée au suivi de la semaine. Nous pensons qu'il serait intéressant d'insister sur cet aspect pour les projets des années futures.

Nous avons aussi remarqué que nous manquions de connaissances pour la mise en place de structures agiles, et qu'il aurait été intéressant que nous essayions de mettre en place des schémas classiques et connus plutôt que de mettre en place des techniques approximatives qui n'étaient pas vraiment aboutissantes. Cela nous a en effet permis de mettre nos compétences d'organisation à l'épreuve, mais nous n'avons pas beaucoup plus d'idées sur le genre de structures mises en place dans les entreprises. Cela signifie que nous avons eu du mal à donner des rôles précis à chacun, du scrum-master au développeur en passant par le testeur, car nous cherchions en permanence une meilleure méthode pour la répartition du travail.

Finalement, bien que nous ayons réussi à mettre en place une gestion d'équipe nous ayant permis de réaliser ce projet de manière satisfaisante, nous avons toujours l'impression que l'aspect lié à la gestion du projet était incomplète, avec des éléments manquants que nous aurions dû mettre en place, sans pour autant trouver ces éléments de gestion manquant. C'est pourquoi nous aurions voulu en apprendre plus sur les méthodes agiles en amont du projet, sur les outils à utiliser pour mettre en place une coordination efficace et sur les "best-practice" des méthodes agiles, afin de suivre des structures de gestion éprouvées et établies.

V. Conclusion

Nous sommes donc, pour conclure, satisfaits de nos travaux et de notre adaptation face à la responsabilité de la gestion de projet. Nos méthodes nous ont permis de mener nos travaux à bien, malgré quelques erreurs du compilateur. Nous avons eu l'opportunité de découvrir la mise en place de méthodes agiles, ce qui constitue une expérience intéressante et enrichissante, mais nous avons aussi dû gérer le travail en groupe dans le contexte du télétravail et gérer des dates de rendus sur un intervalle de temps serré tout en délivrant un travail de qualité, attendu d'un étudiant ingénieur à Grenoble INP - Ensimag. Cette expérience est donc très importante pour chacun d'entre nous, et restera une de nos références en terme de réalisation de projet.

VI. Annexe : la Charte d'Équipe



Projet GL 2021: Charte d'équipe

*Équipe n°31 : Antoine Briançon, Axel Glorvigen,
Thibault Launay, Maxime Martin et Pierre Pocreau*

La charte ci-présente, rédigée, lue et approuvée par tous les membres de l'équipe 31 du Projet GL 2021, présente les règles de travail, d'organisation et de communication que tous les membres de l'équipe 31 s'engagent à respecter sur la durée du projet. Cette charte vise à assurer une bonne entente et une véritable cohésion au sein de l'équipe sur toute la durée du projet. De plus, cette charte propose une analyse des forces et faiblesses de l'équipe à partir de l'analyse des fiches personnelles et d'une matrice SWOT.

Les valeurs communes :

Nous, l'équipe 31, nous engageons, pour ce Projet GL 2021, à suivre les règles et les valeurs communes suivantes :

- Pouvoir parler de tout, selon les positions et les points de vue de chacun, dans le respect et l'écoute de l'autre.
- Faire confiance à tous les membres de l'équipe.
- Ne pas dissimuler les problèmes rencontrés mais les partager avec les autres membres de l'équipe pour trouver des solutions.
- Faire preuve d'initiatives, tout en informant les autres membres de l'équipe sur les recherches et avancements effectués.
- Avoir le droit à l'erreur, et accepter que des solutions ne soient pas parfaites.
- Ne pas hésiter à partager toute information pouvant être utile à l'équipe et à l'avancement du projet.
- Être ponctuel et présent aux réunions d'équipe quotidiennes.

- Respecter les temps de repos (soirs, nuits, week-end) si aucun accord n'a été passé pour travailler sur ces périodes décalées.
- Accepter d'avoir tort, et privilégier le débat à toute autre forme de communication non appropriée dans le cadre de ce projet.

Rôles de responsabilités dans l'équipe :

Nous essayons d'appliquer un fonctionnement en méthodes agiles, et il n'y a donc pas de rôles définis et fixes pour chaque membre de l'équipe. Cependant, nous avons identifié plusieurs rôles essentiels que chacun devra prendre au moins une fois lors de la réalisation de ce projet. Nous avons notamment :

- Le développeur, qui étudie les spécifications données, discute des solutions d'implémentation possibles et s'occupe de modifier le code.
- Le testeur, qui se renseigne sur les spécifications et crée des tests valides et invalides pour certaines parties du projet.
- Le scrum-master, qui gère l'avancement du projet, la répartition des tâches, le suivi des documents administratifs et assure la communication avec le professeur référent.
- Le relecteur de code, qui relit le code du développeur et vérifie que les tests proposés par le testeur sont cohérents et fonctionnent.

Ces rôles sont tournants au sein de l'équipe et, conformément aux méthodes agiles, chacun peut choisir, pour un sprint ou une tâche, de prendre un certain rôle, ce qui permet à chacun d'acquérir de l'expérience dans les domaines qui ne lui sont pas familiers, tout en gardant de l'aisance dans les tâches bien connues.

Communication au sein de l'équipe :

La communication au sein de l'équipe est primordiale. Dans le contexte actuel, nous privilégions les outils de communications à distance, tels que Discord, Messenger ou encore Zoom. Les étudiants vivant à Grenoble, ou partageant la même colocation, profitent de réunions et d'interactions en présentiel, ce qui facilite les échanges à propos du projet. Afin de suivre un schéma se rapprochant du fonctionnement des méthodes agiles, nous organisons au moins une réunion par jour afin d'attribuer les tâches, de discuter des soucis rencontrés et d'avoir un suivi précis de l'avancement du projet.

Nous n'avons pas de règles particulières liées à l'encadrement des réunions, hormis la ponctualité et le fait de parler des problèmes rencontrés dès que ces derniers sont identifiés.

Les réunions ne sont pas formelles. Nous ne rédigeons pas de compte-rendus de réunion, mais nous utilisons les outils de planification de la plateforme Gitlab afin de noter et de suivre l'avancement des différentes tâches identifiées.

Nous utilisons l'outil Git pour la mise en commun de nos travaux. Nous essayons de créer une branche par domaine ou partie de projet, et une attention

particulière est portée à la répartition des branches, ainsi que sur les fichiers modifiés, dans le but de limiter les conflits.

Nous pensons qu'aucun conflit n'aura lieu lors de ce projet. Cependant, si nous nous trouvions dans une situation de conflit, nous privilégierions la diplomatie et le débat afin de désamorcer les tensions. Nous avons confiance en chacun et en la conscience que chacun a de ses responsabilités dans ce projet.

Compétences, points forts et points faibles de l'équipe :

À partir des fiches personnelles remplies par chaque membre de l'équipe (cf annexe), on peut conclure que les compétences techniques de l'équipe sont très solides, et que des qualités comme la persévérance, la débrouillardise ou encore l'esprit d'équipe sont clairement des points forts de l'équipe. Ces compétences ne sont pas négligeables et seront un atout essentiel pour la réussite de ce projet.

Au contraire, les points faibles de l'équipe sont plutôt liés aux compétences de gestion d'équipe de mise en place de méthodes agiles, du fait du manque d'expérience de la plupart des membres de l'équipe. L'organisation et la planification sont notamment des points à travailler, et peuvent considérer des points critiques à maîtriser pour la réalisation de ce projet.

Matrice SWOT :

