

Bilan individuel

Adrien Kaufman, G27

Sur les 5 compétences présentées, je me suis réellement senti progresser sur 2 compétences : *travailler en mode projet* (point 2) et *mettre en œuvre des processus de validation* (point 3).

Travailler en mode projet

Cette compétence était complètement nouvelle pour moi, bien que j'ai eu vent des différentes méthodes de gestion de projet auparavant.

La mise en place du *planning* via l'élaboration d'incrément était dans les grandes lignes une étape assez facile pour notre équipe. Cependant, le majeur problème auquel j'ai (nous avons) été confronté est le fait qu'il est bien plus efficace de réaliser cette étape lorsque l'on a une bonne vue d'ensemble sur le projet. Comme je suis un peu lent, j'aurais dû concentrer bien plus de temps sur la compréhension du sujet (entre le polycopié, les vidéos, les séances et les problèmes de configuration machine j'étais un peu perdu au début), ce qui -je pense- aurait été plus productif, quitte à prendre 2 jours de "retard" en début de projet. Toutefois le pré-découpage (en étapes A,B,C et hello world, sans-objet, complet) a aidé à la construction du planning.

Le *planning* prévisionnel a été globalement respecté, bien plus que ce à quoi je m'attendais (ce qui a été dit en séance de gestion de projet me laissait penser qu'il fallait retoucher le *planning* tous les 4 jours), avec quelques fois un jour de retard, mais la structure globale du *planning* n'a pas changé (pas de chamboulement).

La gestion des rendus intermédiaires m'a aussi quelque peu perturbé : contrairement aux autres projets que j'ai faits, je devais ici me soucier de la qualité du code en milieu de projet. Par exemple au niveau des interfaces : souvent on crée des programmes un peu exotiques temporaires dont on est le seul à comprendre le fonctionnement, et comme inconsciemment l'objectif en cours est le rendu final, j'ai un été légèrement destabilisé et ai dû faire le ménage de ces programmes exotiques intermédiaires. En fait la gestion des *deadlines* intermédiaires m'a forcé à ne pas faire de code intermédiaire bâclé : presque 100% du code que j'ai écrit est dans le rendu final (pas de code supprimé parce que "pas beau").

En ce qui concerne les outils de développement de projet, Excel m'a suffi. J'ai un peu été submergé par les nouveaux outils, j'ai été dépaycé un peu trop vite ; l'utilisation d'Excel pour l'affichage des incrément bien que simpliste m'a permis d'appréhender plus simplement la gestion de projet.

Enfin, j'ai l'impression qu'il m'a fallu un plus d'une semaine pour avoir mes repères finaux dans le projet, et ce n'est qu'au milieu du développement du déca complet que je me sentais au mieux d'avoir une organisation très souple : ce

n'est qu'en fin de projet que j'aurais pu modifier mon environnement de travail (utiliser un nouvel outil, un nouveau logiciel, etc.) de manière optimale (je me trompe peut-être, c'est mon ressenti). Je suis passé de Éclipse à VS Code en cours de projet et j'ai tardivement mis la main sur JaCoCo (outil pourtant simple mais que je redoutais, à cause du manque de repères probablement).

Mettre en œuvre des processus de validation

Cette étape était à moitié nouvelle pour moi car j'avais déjà réalisé des processus de validation mais à mon gré, ici la validation était règlementée. J'étais chargé de cette étape tout au long du projet.

C'est la première fois que je mets vraiment les mains dans le *bash*, et cela m'a beaucoup aidé à titre personnel (je me demande comment j'ai pu faire sans avant...). J'ai pris assez de plaisir à développer la structure automatique pour la gestion des tests.

À cause de problèmes personnels (je ne veux pas parler en son nom) un membre de notre équipe a eu des difficultés pour suivre le projet, je me suis donc retrouvé plus ou moins seul sur la gestion des tests¹ : les trois développeurs avançaient donc "trop" vite et même lorsqu'ils ajoutaient eux-même des tests il a été assez difficile d'en créer beaucoup. Je suis mitigé quand à la gestion de la base de tests : d'un côté le développement du compilateur avançait légèrement plus vite que le développement des tests donc on avait la sensation d'avancer vite, dans la bonne direction, et d'un autre côté il n'y avait pas assez de tests. J'avais donc l'impression que le projet avançait vite comme s'il était bâclé et pourtant je n'avais pas la sensation qu'il était bâclé (un peu comme quand on finit un contrôle trop tôt, on se dit qu'on a râté quelque chose mais on a la sensation d'avoir bien fait).

Je pense que c'était une bonne idée qu'il y ait une personne (en l'occurrence moi) chargée de la gestion de validation : ceci permettait aux autres de ne pas se focaliser dessus, et je leur disais les résultats (quels tests ont été créés, quels tests ne passent plus,...).

Ce qui a donc été le plus problématique dans cette compétence est la gestion du nombre de tests ; j'ai dû passer trop de temps sur des aspects subsidiaires, et puis je n'avais pas utilisé JaCoCo (ce qui aurait été assez pratique...).

Conclusion

En fin de compte, ce projet m'a permis d'appréhender la gestion de projet, notamment l'étape de validation. Un point qui ressort particulièrement est la gestion des scripts (c'est "l'illumination" technique de ce projet pour moi) que j'emploierai à l'avenir. Le principal point négatif que je ressens pour ce projet est la mise en place : j'ai mis beaucoup de temps (plus que d'habitude) à avoir mes repères, donc mon confort et ma productivité ont baissé.

Le fait de ne pas avoir de vision globale m'a frustré pendant une bonne partie du projet.

En somme, il est sûrement plus efficace pour une équipe de se spécialiser, mais j'ai l'impression qu'individuellement il me convient davantage de privilégier la vision globale (avant de me spécialiser).

1. je n'étais pas seul à écrire tous les tests tout le temps, surtout en fin de projet