

UNIVERSITÉ DU QUÉBEC À MONTRÉAL



TP4-A: RAPPORT DE RÉTROSPECTIVE GLOBALE

TRAVAIL PRÉSENTÉ À M. PHILIPPE DESCHÊNES
COMME EXIGENCE PARTIELLE DU COURS INF6150

PAR
OLIVIER BOHIGAS
MATHIEU DES LAURIERS
JÉRÉMIE GOUR
JULIEN MELANÇON
DANIEL ROSCA

UNIVERSITÉ DU QUÉBEC À MONTRÉAL
DÉPARTEMENT D'INFORMATIQUE
ÉTÉ 2021

Nous avons, durant toute la durée du projet, profité d'une très bonne communication au sein de l'équipe. En effet, notre utilisation active du serveur Discord permettait de voir les traces d'avancement du projet, les obstacles et les risques encourus. Le gabarit d'activités et notre suivi via la section Issues du GitLab permettaient aussi très rapidement et facilement d'identifier les différentes tâches en cours et qui travaillaient sur ces tâches. De plus, nous avons su profiter de l'expertise de quelques membres sur certains éléments plus complexes à résoudre ce qui nous a permis de diminuer le temps passé sur les plus gros problèmes. Dernièrement, le travail en binôme permettait à chaque petit groupe de profiter des forces de chacun pour pallier le manque d'expérience dans certains domaines techniques. Cela a réduit les erreurs et la charge mentale sur les diverses fonctionnalités à implémenter, augmenté la productivité et la motivation lors du développement et diminué la difficulté de certaines parties plus complexes à développer.

Malgré toutes les bonnes pratiques mentionnées ci-haut, nous avons eu plus de difficultés avec d'autres ce qui a impacté négativement notre développement. Premièrement, la majorité du travail, à chaque sprint, se faisait plus sur la deuxième moitié du temps alloué que sur la première. Cela faisait donc en sorte que notre marge de manœuvre pour assurer la qualité des sprints était limitée. De plus, le manque de supervision globale portait problème sur l'uniformisation du travail effectué sur les divers modules développés en binôme. Malgré le fait que nous travaillions pratiquement tout le temps en équipe, attribuer une personne dédiée à la revue de code aurait été pertinent. Dernièrement, un problème qui est survenu à toutes les itérations était la sous-estimation du temps d'apprentissage de certaines technologies utilisées. Cela venait constamment bouleverser nos estimations puisque la réalisation de certains aspects qui semblaient élémentaires était souvent plus longue que prévu.

Nous avons soulevé 4 mauvaises pratiques qui sont venues affecter la performance globale de l'équipe. Tout d'abord, le manque de planification des horaires dans les itérations faisait en sorte que chacun pouvait travailler à n'importe quel moment de la journée et de la semaine. Cela apportait une certaine lourdeur psychologique puisque nous pouvions constamment être contactés pour une question ou un problème relié au projet. Ensuite, nous avons passé trop de temps sur l'implémentation de fonctionnalités ce qui nous laissait en contrepartie trop peu de temps sur la révision de la qualité de notre travail. Dernièrement, nous faisions trop peu de commentaires sur le code développé et il était donc difficile pour les autres membres de l'équipe de toujours être en mesure de comprendre rapidement et facilement ce qui avait été fait par les autres membres.

Pour conclure, nous avons mis en place un plan d'amélioration afin de corriger notre processus de développement. Premièrement, il est difficile de coordonner les horaires de cinq personnes à chaque semaine, mais réserver du temps à un moment particulier pourrait empêcher les débordements et permettre de naviguer avec moins d'incertitude. Deuxièmement, quitte à devoir reconduire la finalisation de certains *user stories* à l'itération suivante, il faudrait s'assurer que celles implémentées réussissent nos tests d'acceptance. Troisièmement, en rapport avec la qualité, mais aussi avec la communication, il faudrait que l'ensemble des développeurs participent activement à rendre le code plus lisible et clair. Cela permettrait de ne pas perdre de temps à comprendre comment les différentes parties de l'application ont été réalisées. Dernièrement, le design était la partie la plus coûteuse du développement. Avoir un expert ou quelqu'un qui développe son expertise dans ce domaine aurait permis d'augmenter la vélocité du projet.