

GLO-4008/GLO-7008 - Travail Pratique 2

Questionnaire

Equipe 4

Ludovic HU - 537 174 920

Bappi RAHAMAN - 537 046 96

Allan TARCY - 537 169 500

1. Combien d'années d'expérience avez-vous dans le développement de logiciels?

Nous sommes tous étudiants en 4 ou 5 ème année d'études supérieures. Donc nous n'avons pas encore d'expérience professionnelle.

2. Avez-vous déjà effectué une décomposition d'application en microservices? (Oui / Non)

Aucune, sauf celle réalisée en séance de laboratoire.

Expérience de la tâche

3. Comment évalueriez-vous la difficulté de décomposer la première application en utilisant le protocole manuel (Très facile / Facile / Neutre / Difficile / Très difficile)? Pourquoi?

Neutre, la réalisation n'était pas particulièrement difficile, cependant, on aurait aimé l'avoir fait une fois en lab au préalable. La décomposition nous a un peu porté à confusion lorsqu'il s'agissait de séparer des classes liées par héritage notamment.

4. Sur une échelle de 1 à 5 (1 pas intuitif du tout et 5 très intuitif), à quel point Trouvez-vous le processus de décomposition manuelle intuitif? Pourquoi?

4. Comme expliqué précédemment cela reste plutôt intuitif la plupart du temps, cependant on a eu quelques interrogations.

5. Combien de temps vous a-t-il fallu pour terminer la décomposition manuelle?

Environ 3 heures.

Efficacité

- 6. Sur une échelle de 1 à 5 (1 pas efficace du tout et 5 très efficace), comment évalueriez-vous l'efficacité de l'approche de décomposition manuelle? Pourquoi?**

2. Il est assez chronophage de passer chaque classe une à une et d'étudier manuellement toutes leurs dépendances. Le temps d'exploration augmente énormément avec des applications plus conséquentes.

- 7. Veuillez décrire les défis auxquels vous avez été confronté lors du processus de décomposition manuelle**

L'architecture de l'application était inconnue, nous avons déjà rencontré quelques exemples de DDD, mais pas ce type d'architecture. On se demandait si toutes les règles d'assurance qualité étaient bien respectées par moment. De ce fait, la décomposition était un peu plus délicate car parfois, certaines classes possèdent des dépendances très génériques.

- 8. Quelle approche de décomposition automatique avez-vous trouvée la plus efficace et pourquoi?**

MSExtractor semblait bien adapté sur cette petite application. La décomposition était étonnamment proche de celle que l'on a considérée. Cette solution était la plus intelligible.

- 9. Les outils automatisés ont-ils fourni des décompositions claires et utiles (Oui / Non)? Pourquoi?**

En dehors de MSExtractor, les solutions proposées paraissaient assez aléatoires. Le nombre de microservices générés était trop grand pour la taille des applications.

- 10. Comment compareriez-vous le temps et les efforts nécessaires pour décomposer l'application à l'aide d'outils automatisés par rapport à l'approche manuelle?**

Pour Mono2micro, on peut hésiter avec la décomposition manuelle car le processus est un peu laborieux. Pour les autres applications, la procédure est plus rapide et plus simple.

- 11. Dans quelle mesure êtes-vous confiant dans la maintenabilité et la scalabilité des microservices créés par chaque approche?**

Aucune approche ne semble prête à recréer un système fonctionnel. Il est donc difficilement envisageable de maintenir ces microservices.

12. Quel type de ressources ou de support supplémentaires auraient facilité le processus de décomposition?

Un développeur expérimenté aurait été très utile lors de cette décomposition.

13. Globalement, quelle approche préféreriez-vous pour décomposer des applications en microservices et pourquoi?

Pour ce cas précis, la solution proposée par MSExtractor était la plus intéressante. Les solutions obtenues permettaient vraiment d'obtenir une base de travail.

14. Dans quels scénarios envisagez-vous que les outils de décomposition automatisés seraient les plus bénéfiques?

Pour retrouver les contextes délimités, les approches automatiques seraient très performantes.

15. Y a-t-il des défis techniques spécifiques avec les outils automatisés que vous souhaitez souligner?

Le problème repose sur le fait que ces approches ne permettent pas de simuler le processus de réflexion complexe d'un architecte logiciel. Pour l'instant, il faut beaucoup compter sur la qualité du code existant pour obtenir une décomposition satisfaisante.

16. Dans quelle mesure les outils automatisés que vous avez utilisés délimitent-ils efficacement les « bounded contexts » au sein d'une architecture microservices?

Tant que les objets sont correctement nommés, les outils automatisés réalisant une analyse sémantique réussissent à regrouper les concepts individuellement. On observe par contre une difficulté à retrouver des relations plus intimes entre des classes sémantiquement éloignées.

Satisfaction de l'utilisateur

- 17. Dans quelle mesure êtes-vous satisfait du résultat du processus de décomposition manuelle par rapport à l'automatisation? (Très insatisfait / Insatisfait / Neutre / Satisfait / Très satisfait) (pourquoi?)**

Satisfait. L'analyse que nous avons porté sur le code et les connaissances préalables sur l'architecture logicielle, nous ont permis d'émettre des hypothèses plus rationnelles que les fonctions objectives n'arrivent pas toujours à saisir.

- 18. Quels aspects des outils automatisés pensez-vous avoir besoin d'amélioration?**

Les outils automatisés devront nécessairement posséder un processus de réflexion similaire aux architectes logiciels. Les considérations à apporter sont nombreuses, et une analyse sémantique même complexe ne peut pas pleinement résoudre le problème des monolithes.

- 19. Envisageriez-vous d'utiliser les outils automatisés de décomposition pour des projets futurs? (Certainement pas / Peu probable / Peut-être / Probable / Certainement)**

Peu probable. Cependant, pour ceux qui sont confrontés à ce genre de problématiques, les gains potentiels sont réels.

Feedback ouvert

- 20. Veuillez fournir tout commentaire ou suggestion supplémentaire concernant les processus de décomposition que vous avez essayé**

Nous avons déjà mentionné les difficultés rencontrées, mais nous pouvons rappeler les quelques interrogations que nous avons sur la façon de séparer les interfaces dans les différents services.