Proposition de projet - Équipe 1

* Quel(s) projet(s) en TypeScript visez-vous pour votre étude empirique?

Pour notre étude empirique sur TypeScript, nous avons sélectionné une liste variée et intéressante de projets. Ces projets sont pertinents pour une étude sur les bogues dans des applications TypeScript. Voici un bref aperçu de quelques projets notables de notre liste :

|  |  |
| --- | --- |
| **Nom du Projet** | **Emplacement sur GitHub** |
| antvis\_\_G2 | https://github.com/antvis/G2 |
| antvis\_\_G6 | https://github.com/antvis/G6 |
| chartist-js\_\_chartist | https://github.com/chartist-js/chartist |
| GeekyAnts\_\_NativeBase | https://github.com/GeekyAnts/NativeBase |
| infinitered\_\_ignite | https://github.com/infinitered/ignite |
| jacomyal\_\_sigma.js | https://github.com/jacomyal/sigma.js |
| nhn\_\_tui.editor | https://github.com/nhn/tui.editor |
| peers\_\_peerjs | https://github.com/peers/peerjs |

* Quel genre d'étude empirique proposez-vous?

Nous envisageons de réaliser une étude empirique sous la forme d'une expérience de réplication, qui a pour but d'évaluer comment la qualité du code influence la fréquence des bogues dans des projets développés en TypeScript.

* Quelles sont les hypothèses ou les questions de recherche?

Notre étude vise à mesurer l'effet de diverses métriques de qualité sur la fréquence et la sévérité des bogues au sein de projets logiciels. Nous nous baserons sur les hypothèses suivantes pour orienter notre recherche :

1. Hypothèse 1 : Une grande complexité cyclomatique au niveau des méthodes augmente la présence de bogues. Cette hypothèse s'appuie sur le principe que plus un bloc de code est complexe à comprendre et à tester, plus il est susceptible de contenir des erreurs non détectées.
2. Hypothèse 2 : Une grande quantité de lignes de code, que ce soit au niveau des méthodes ou des classes, augmente la présence de bogues. Cette relation suppose que l'augmentation de la quantité de code augmente les chances d'introduction d'erreurs.

* Quels sont les informations (métriques?) à obtenir dans votre analyse pour faire l'étude?

Pour réaliser notre étude empirique, il est essentiel de collecter et d'analyser un ensemble spécifique de métriques qui seront quantifiées à l'aide de Pharo. Les métriques clés à considérer sont :

* Complexité Cyclomatique (CC) : qui mesure le nombre de chemins d'exécution possibles dans un segment de code, reflétant sa complexité logique. La complexité cyclomatique sera analysée seulement au niveau des méthodes, offrant une évaluation détaillée de l'impact de la complexité sur la qualité du code.
* Lignes de Code (LOC) : qui compte le nombre total de lignes de code effectives, en excluant les commentaires et les lignes vides. Elle sera analysée à la fois au niveau de chaque méthode et de chaque classe, permettant une analyse approfondie de comment la quantité de code influence la fréquence des bogues.

Ces métriques seront extraites des fichiers identifiés comme contenant des bogues, grâce à l'utilisation de l'outil SZZ.

* Quels sont les rôles (d'expertises, p.ex. coordination, en TypeScript, en Pharo/Moose, en analyse, etc.)?
* Pinto Carlos Jose : Expertise en programmation et en coordination.

Carlos mettra à profit son expertise pour mettre en place un pipeline d'analyse de code utilisant SZZ. Il sera chargé de l'intégration du code et participera également à la coordination des activités de l'équipe afin d'assurer une progression cohérente du projet.

* Jouhaina Nasri : Expertise en analyse des métriques.

Jouhaina prendra en charge l'analyse des métriques et calculera la corrélation entre ces métriques pour valider les hypothèses de l'étude.

* Mohammed Boutahar : Expertise en Pharo.

Mohammed se concentrera sur l'utilisation de Pharo pour développer un code capable de générer les métriques nécessaires à notre étude.

Les contributions de chaque membre de l'équipe seront cruciales pour le succès de cette étude empirique.

* Y a-t-il des problèmes dans l'équipe déjà et comment avez-vous essayé de les régler (selon le texte sur les parasites et les mollassons)?

Jusqu'à présent, aucun problème n'a été signalé au sein de notre équipe, et la collaboration se déroule dans une atmosphère harmonieuse. Pour assurer une coopération efficace, nous avons adopté des stratégies de communication transparente, de respect mutuel et de confiance, , inspirées des principes énoncés dans le texte sur les parasites et les mollassons. Dans l'éventualité où des difficultés surgiraient, nous sommes prêts à les gérer en appliquant les méthodes suggérées afin de préserver un climat de travail à la fois positif et productif.