

Stratégie de mise en place de revues de code

Date : 10 avril 2015

Format : Quickie

Speakers : David Wursteisen, Soat

Ce quickie fait part d'un retour d'expérience sur la mise en place de revue de code au sein d'une équipe de 4 développeurs

Organisation avant Scrum :

Application comportant plusieurs briques.

Inconsciemment, chaque développeur travaille sur sa brique. Chacun est expert de sa brique.

Depuis la mise en place de Scrum :

La personne disponible travaille sur la brique en question.

Une remarque revient régulièrement : « Mais qu'est-ce que c'est que ce code ? »

En effet, chacun intervient désormais sur le code des autres.

D'autres problèmes sont remontés :

- Deux écoles dans l'équipe : ceux qui commencent leurs conditions par un *if* et ceux qui préfèrent le *if not* pour sortir plus tôt de la méthode
- Code fonctionnellement identique mais codé différemment
-

Lors des Sprint, la qualité du code est pointée du doigt.

Moyen d'y remédier : les revues de code. C'est ce qui est fait chez Google, GitHub ...

Essai de mise en place d'une revue de code systématique

Mais en fin de Sprint, aucune revue n'a été réalisée

Rétrospective : toujours pas fait de revue de code. Peut-être trop ambitieux de les systématiser ? Mise en place d'une revue de code nécessaire => fonctionne toujours pas.

Essai avec l'utilisation de 2 post-it revues de code rattachés aux tâches => plutôt bien marché. Tant que la revue de code n'est pas faite, on ne passe pas les tâches à l'étape Terminée.

Comment faire les revues de code ?

- Revue par mail : très mauvaise idée car trop fastidieux
- Utilisation de Stash : on peut proposer des modifications + des commentaires => très bien, mais nécessite Git.
- Le développeur le revoit avec une autre personne de l'équipe. En cas de désaccord, le problème est soumis au reste de l'équipe. Un consensus doit s'établir.

Quels sont les impacts sur le code ?

- StringBuilder non nécessaire
- Frameworks de log (type SLF4J) permettant de passer des paramètres sans faire isDebugEnabled
- Log d'erreur + exception levée

Ces revues permettent d'uniformiser le code et les pratiques de dev.

Quels sont les impacts dans l'équipe ?

- Du binôme s'installe entre les devs

Cette pratique peut disparaître sans que ce soit volontaire.

Pour cela, on peut inclure la revue de code dans la Definition of Done.