

Ratez ses revues de code en 5 leçons

Speaker : Michel Domenjoud (Octo)

Format : Quickie

Date : 22 avril 2016

Slides : <http://fr.slideshare.net/mdomenjoud/ratez-vos-revue-de-code-en-5-lecons-devoxx-fr-2016>

Première pratique à mettre en œuvre dans une équipe.

Constat : des projets se sont très bien passés. Coïncidence, la revue de code se pratiquait. D'autres équipes avaient peur de changer leur code et avaient peur de mettre en prod. Pas de revue de code ?



Dans l'histoire de l'informatique, la revue de code est la 1^{ière} pratique collaborative (années 60).

Objectif :

- ⇒ Détecter des défauts. Etudes : 65% des défauts peuvent être évités. 1h de revue de code = 4h de debug économisée
- ⇒ Bénéfices conséquents
 - Partager les standards
 - Partager les connaissances

3 types de revue de code :

- ⇒ Revue par un pair
- ⇒ Revue collective
- ⇒ Pair programming

3 histoires

Fail#1 : On croule toujours sous les bugs !

Site de réservation de billets de train en ligne.

Fonctionnalité complexe développée en 2 semaines mais bien pensée. Revue de code OK d'après Benoît.

Passage en prod => 15 jours + tard gros bug qui crash.

Pourquoi la revue n'a pas permis d'identifier le bug ?
Elle a été réalisée trop rapidement.

Bonnes pratiques :

- Prendre le temps, à un rythme soutenable. Pas plus de 300 lignes de code pour 1h.
Pas plus de 1h30
- Demandez une revue dès que possible
- Préparer la revue via une lecture à tête reposée

Fail#2 Mon équipe ne progresse pas

Coache une équipe. Les mêmes erreurs reviennent régulièrement.
Raisons : ne se parlent pas, corriger à la place de l'auteur

Bonnes pratiques :

- Echangez de vive voix pendant les revues
- L'auteur corrige, en binôme si nécessaire
- Ecrivez vos standards. Les faire évoluer

Fail #3 : On se tape dessus

Lors d'une revue de code collective, la tournure du débat s'est mal passée. Un développeur est sorti en colère.

Bonnes pratiques :

- Critiquez le code, pas le développeur
- Appuyez vous sur les principes Egoless Programming
- Ce n'est pas le moment pour avoir des discussions à rallonge
- Trouvez des compromis et en faire un standard.

Ces 3 formats de revue sont complémentaires

Trouvez le ou les formats qui conviennent et adaptez le au besoin :

- Pair programming : Démarrer une US, Code Legacy, Debug, Apprendre
- Revue par un pair : tout le temps

Livre [Culture Code](#) gratuit sur le site d'Octo