Encadrement

* Jenkins

|  |
| --- |
| Pourquoi surveille-t-on ça ? |
| * Pour vérifier si le code compile/teste/… sur une plateforme indépendante des machines de devs. * Pour vérifier que le code s’intègre bien. |
| Comment obtenir le feedback/les indicateurs ? |
| * Suivre la partie « Intégrer mon code » du guide du développeur.   + Se connecter à ce lien : <http://vps575474.ovh.net:8080/>.   + Lancer le build   + … |
| Comment agir sur les feedback/les indicateurs ? |
| * Si le problème vient de la compilation / tests, corriger en local. * Faire un « mvn install » avant de commiter pour vérifier la compilation. * Nommer le commit clairement : « [correction CI] : … ». |

* Sonar

|  |
| --- |
| Pourquoi surveille-t-on ça ? |
| * Pour vérifier que les règles implémentées dans sonar sont respectées. * Permet de détecter d’éventuels problèmes lié à de mauvaises pratiques par exemple. |
| Comment obtenir le feedback/les indicateurs ? |
| En suivant ce lien : <http://vps575474.ovh.net:9000/>. |
| Comment agir sur les feedback/les indicateurs ? |
| * Les différents problèmes relevés par sonar sont à corriger en local. * Sonar détecte les bugs et les vulnérabilités à corriger. * Analyser le coverage des tests unitaires. |

* Git flow ?

|  |
| --- |
| Pourquoi surveille-t-on ça ? |
| * Pour empêcher de laisser n’importe quel commit passer sur la branche de développement. * Permettre de faire des revus de code. * Faire les nouveaux développements sur des branches feature pour partitionner le code. |
| Comment obtenir le feedback/les indicateurs ? |
| En suivant ce lien : <https://github.com/loicmolina/gamestore-api/pulls>. |
| Comment agir sur les feedback/les indicateurs ? |
| * Une fois les nouveaux développements terminés sur la branche de feature, on merge sur la develop. * Pour une livraison, cela ce produit sur la release et la version finale sur le master. |

* Outils Github: L’outils « Projects/Gamestore » avec le template Kanban.

|  |
| --- |
| Pourquoi surveille-t-on ça ? |
| Pour vérifier l’avancement du projet selon les différentes tâches à effectuer : To do / In progress / Done. |
| Comment obtenir le feedback/les indicateurs ? |
| En suivant ce lien : <https://github.com/loicmolina/gamestore-api/projects>. |
| Comment agir sur les feedback/les indicateurs ? |
| En travaillant sur les points qui reste en To do ou en in progress. |

* Tests ?

|  |
| --- |
| Pourquoi surveille-t-on ça ? |
| * Les tests permettent de détecter les régressions en cas de modification du code. * Une bonne couverture de ceux-ci permet de s’assurer que le code fait bien ce que l’on veut qu’il fasse. |
| Comment obtenir le feedback/les indicateurs ? |
| * En local : JUnit permet de mettre en place les tests et savoir s’ils passent ou pas. * Sur Jenkins : permet de tester indépendamment de son environnement. * Sur sonar : Permet de regarder le coverage, la structure des tests … |
| Comment agir sur les feedback/les indicateurs ? |
| * Corriger les tests qui ne passent pas. * S’assurer d’un coverage suffisant. |