

L'INTÉGRATION CONTINUE, UN ENJEU POUR LA QUALITÉ

CODE STAGE : F28040

OBJECTIFS

Maîtriser les phases de mise en oeuvre d'une intégration continue

Maîtriser les maillons d'une chaîne d'intégration continue

Automatiser la construction de vos applications

Mettre en place un tableau de bord de vos intégrations

Générer des rapports liés à l'activité de développement et à la qualité

Valider le respect des normes de développement par les développeurs via Sonar

DURÉE

3 jours

PUBLIC

Analyste

Chef de projet

Concepteur

Développeur débutant

PRÉ-REQUIS

Avoir une compréhension d'un langage Java / C++ / .NET et Web

PROGRAMME

Chapitre 1 : Introduction

Comprendre la notion de chaîne d'intégration continue

Les étapes de mise en oeuvre

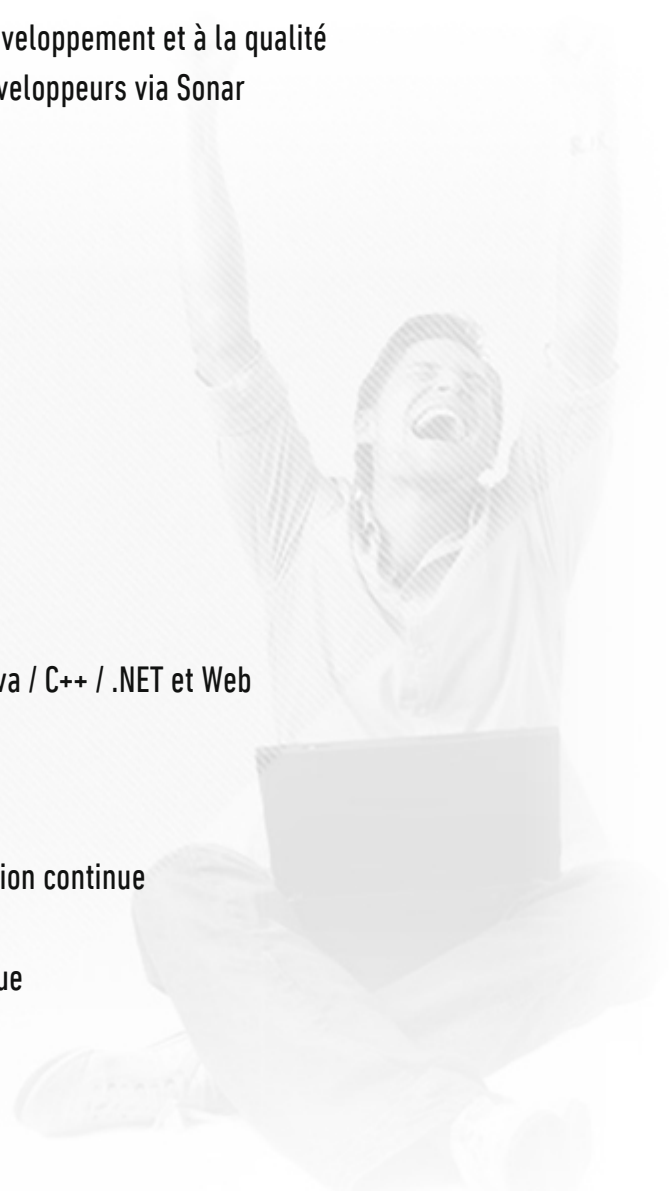
Rôle de la plateforme d'Intégration continue

La place des tests unitaires / intégration

l'acceptance en continue

Le déploiement en continue

Intégration continue et Agilité



Robustesse, Maintenabilité

Chapitre 2 : Le Rôle d'un outil de build

Définition et comparatif

Conventions

Organisation

Mise en place

Complément E-learning sur Maven / Gradle / Ant / MSBuild / Make. Les stagiaires feront une petite présentation d'une dizaine de minutes qui devra résumer les vidéos.

Exemple d'exercice : À travers différents projets réalisés en Java / C++ / .Net sur des Frameworks variés (déjà codés) analyse et optimisation du fichier POM.

Chapitre 3 : Le rôle du gestionnaire de sources (Git, SVN, CVS)

Partager son code de manière constructive

Notion de dépôt local / centralisé

Workflows Git

Complément E-learning sur Git et SVN. Les stagiaires feront une petite présentation d'une dizaine de minutes qui devra résumer les vidéos.

Exemple d'exercice : À travers différents projets mis en place sur GitHub, Bitbucket et GitLAB analyse des informations portées par GIT.

Chapitre 4 : Le rôle des tests

Valider ses tests unitaires

Réaliser et valider ses tests d'intégration

La couverture de code

Complément E-learning sur Junit / Nunit / Boost Test et les outils de charge.

Exemple d'exercice : À travers différents projets réalisés en C++ / .NET / Java sur des Frameworks variés (déjà codés) validation de la cohérence et de la pertinence des tests.

Chapitre 5 : L'analyse qualité du code

La qualité de code dans le processus de build

Rapport sur la complexité du code : Cobertura, JaCoco

Rapport sur les tâches ouvertes (FIXME, TODO,)

Installation de Sonar Qube

Intégration avec l'outil Sonar

Les 7 axes de Qualité logicielle

Analyse de code via Sonar Runner

Analyse de code via plugin Eclipse

Paramétrage de règles spécifiques

Profils de règle

Exemple d'exercice : À travers différents projets réalisés sur des Frameworks variés (déjà codés) mise en place d'un Sonar afin qu'il analyse chaque projet.

Chapitre 6 : Automatiser ses builds avec Jenkins

Valeur ajoutée d'un serveur d'Intégration Continue

Installation Jenkins

Configuration JDK et Maven

Création premier Job

Lancement premier Job

Les rapports

Tableau de bord de la configuration Jenkins

Complément d'E-learning sur Hudson et Jenkins. Les stagiaires feront une petite présentation d'une dizaine de minutes qui devra résumer les vidéos.

Exemple d'exercice : À travers différents projets réalisés sur des Frameworks variés (déjà codés) mise en place d'un Jenkins afin qu'il automatise le cycle d'IC

