PROGRAMME DE FORMATION #GIT-1

PRÉSENTATION

Git est un outil indispensable qui permet de travailler en équipe sur un projet versionné. Lorsqu’il est mis en place, sa maîtrise est nécessaire à toute évolution et bon maintien du projet. Venez découvrir comment cette formation répond à ces besoins.

OBJECTIFS DE LA FORMATION

**Comprendre la philosophie Git et l'intérêt du versioning**

* Travailler sur un projet en équipe versionné
* Maîtriser Git en lignes de commandes
* Améliorer la résolution de bugs grâce à Git

**Préparer son environnement**

* Installer et configurer Git
* Création d'un dépôt Git local

**Comprendre le fonctionnement de Git**

* Les différents termes Git
* Structure d'un dépôt Git
* Les différents objets gérés par Git

**Apprendre les commandes basiques Git**

* Ajouter des fichiers au staging
* Obtenir des informations sur l'état de la copie de travail
* Consulter les modifications en cours
* Supprimer les modifications courantes
* Versionner l'état courant grâce au commit
* Analyser l'historique
* Afficher le contenu d’un commit spécifique
* Mettre de côté un travail en cours Comprendre le système de branches
* Comprendre le principe de branche
* Apprendre à utiliser les branches

**Apprendre les commandes basiques sur les branches**

* Créer une branche
* Naviguer entre les branches
* Supprimer une branche
* Renommer une branche

**Manipulation avancée des branches par merge et rebase**

* Le merge avec fast-forward
* Le merge sans fast-forward
* Le rebase

**Gérer des conflits entre branches**

* Le cas de conflits simples
* Le cas de conflits complexes impactant plusieurs commits
* Le rebase interactif

**Palier aux erreurs humaines fréquentes**

* L'oubli de changement de branche sur une seconde console ouverte
* Réparer le dernier commit effectué
* Supprimer les derniers commits
* Fixer plusieurs commits en réécrivant l’historique
* Annuler un ancien commit sans réécriture de l’historique
* Retirer des données sensibles de l’historique

**Approfondir les commandes basiques**

* Ajouter des petites portions de fichiers au staging
* Consulter les modifications ajoutées au staging
* Aller plus loin avec git diff
* Aller plus loin dans l’analyse de l’historique
* Tagger une révision
* Améliorer la configuration d’un projet local