

Procédure 2 : Gestion des versions de code

Objectif : Mettre en place un système de gestion des versions de code pour assurer la traçabilité et la collaboration efficace lors du développement d'un projet.

Étapes :

1. Choix d'un système de gestion de versions (SGV) :
 - Sélectionner un SGV adapté aux besoins du projet, tel que Git, SVN ou Mercurial.
2. Initialisation du référentiel (repository) :
 - Créer un référentiel (repository) vide sur une plateforme de gestion de code comme GitHub, GitLab ou Bitbucket.
3. Clonage du référentiel :
 - Cloner le référentiel vide sur votre machine locale en utilisant la commande appropriée fournie par le SGV choisi.
4. Ajout des fichiers au référentiel :
 - Ajouter les fichiers du projet au référentiel en utilisant la commande git add (pour Git) ou équivalent dans le SGV choisi.
5. Validation des modifications :
 - Valider les modifications en ajoutant un message de commit décrivant les changements effectués à l'aide de la commande git commit -m "Message de commit" (pour Git).
6. Synchronisation avec le référentiel distant :
 - Pousser les modifications vers le référentiel distant en utilisant la commande git push pour que les autres membres de l'équipe puissent accéder aux dernières modifications.
7. Gestion des branches :
 - Créer des branches pour les fonctionnalités ou correctifs spécifiques en utilisant la commande git checkout -b nom_de_branche (pour Git).
 - Fusionner les branches une fois les modifications testées et validées à l'aide de la commande git merge.

Résultats attendus :

- Un référentiel de code bien organisé et versionné est créé.

- Les modifications apportées au code sont enregistrées avec des messages de commit descriptifs.
- Les collaborateurs peuvent travailler simultanément sur différentes fonctionnalités sans conflits majeurs.
- Les versions précédentes du code sont accessibles pour référence et récupération.

Remarques :

- Assurez-vous de respecter les bonnes pratiques de gestion de version, telles que l'écriture de messages de commit clairs et concis.
- Familiarisez-vous avec les fonctionnalités avancées du SGV choisi, telles que les rebases et les révisions, pour une gestion plus efficace du code.
- Communiquez régulièrement avec les membres de l'équipe pour coordonner les efforts de développement et résoudre les conflits éventuels.