OSD
OPEN DIGITAL SERVICE

Scannez-moi

WHATSAPP
+213 674 05 38 95

SITE WEB
osdigitalis.com

FICHE FORMATION · OSD

Versionner et collaborer avec Git et GitHub

Suivre l'historique d'un projet, travailler en équipe et contribuer proprement

20 heures | 10 séances | ligne de commande | travail en équipe

Une formation pratique pour **maîtriser Git au quotidien et collaborer sur GitHub** comme dans une équipe de développement : branches, fusions, demandes de fusion, revues de code et bonnes pratiques.

20 h	10 × 2 h	débutant	en équipe
DURÉE	SÉANCES	NIVEAU	FORMAT

Description

Git est utilisé par la quasi-totalité des équipes de développement et d'infrastructure : sans lui, pas de travail à plusieurs, pas de retour arrière fiable, pas d'intégration continue. GitHub ajoute la collaboration : revues de code, suivi des tâches, automatisation.

Les participants travaillent **en équipes sur un vrai dépôt partagé** : ils créent des conflits volontairement, les résolvent, relisent le code des autres et publient des versions. Chaque commande est expliquée avec un schéma de ce qui se passe dans le dépôt.

Public visé

- Étudiants en informatique, dès la 1re année
- Développeurs qui utilisent Git sans le comprendre vraiment
- Administrateurs systèmes qui versionnent scripts et configurations

Prérequis

- Utilisation courante d'un ordinateur
- Bases de la ligne de commande appréciées
- Un compte GitHub gratuit et Git installé

À l'issue de la formation, vous saurez

☐	expliquer le modèle de Git : **dépôt, index, commit, branche** ;
☐	enregistrer, comparer et annuler des modifications sans perdre de travail ;
☐	créer et fusionner des **branches**, résoudre les **conflits** ;
☐	travailler avec un **dépôt distant** : clone, push, pull, fetch ;
☐	collaborer sur **GitHub** : pull requests, revue de code, issues, protection des branches ;
☐	appliquer un **flux de travail** d'équipe et des messages de commit propres.

Méthode pédagogique

- Courts exposés avec **schémas de l'historique**, puis manipulation immédiate.
- Travail **en équipe** sur des dépôts partagés, avec des conflits provoqués volontairement.
- Ligne de commande d'abord, outils graphiques (VS Code, GitHub Desktop) ensuite.
- Bonnes pratiques professionnelles : petits commits, messages clairs, revue systématique.

Programme — 10 séances

1 **Pourquoi versionner** 2 h

Problèmes du travail sans historique, systèmes centralisés et distribués, installation et configuration de Git.

`git config` `historique` `VCS`

2 **Premiers commits** 2 h

Dépôt, répertoire de travail, index et commit ; status, add, commit, log et diff.

`git add` `git commit` `git log`

3 **Voyager dans l'historique et annuler** 2 h

Consulter un ancien état, annuler un fichier ou un commit, restore, revert, reset et ignorer des fichiers.

`restore` `revert` `.gitignore`

4 **Les branches** 2 h

Créer, changer et supprimer des branches, comprendre HEAD, fusion rapide et fusion avec commit.

`branch` `switch` `merge`

5 **Résoudre les conflits** 2 h

Provoquer, lire et résoudre des conflits de fusion, outils de résolution, rebase simple.

`conflits` `rebase` `mergetool`

6 **Dépôts distants et GitHub** 2 h

Créer un dépôt GitHub, clés SSH, clone, push, pull et fetch, branches distantes.

`GitHub` `SSH` `push · pull`

7 **Collaborer avec les pull requests** 2 h

Proposer une modification, revue de code, commentaires, demandes de changement et fusion.

`pull request` `revue` `fork`

8 **Flux de travail en équipe** 2 h

GitHub Flow et Git Flow, protection des branches, conventions de commit, issues et tableaux de projet.

`GitHub Flow` `issues` `conventions`

9 **Versions et outils avancés** 2 h

Étiquettes et releases, stash, cherry-pick, retrouver un commit perdu, introduction à GitHub Actions.

`tags` `stash` `Actions`

10 **Projet d'équipe et évaluation** 2 h

Projet collaboratif complet en équipe sur GitHub, puis évaluation pratique individuelle.

`projet` `évaluation /20`

Évaluation et validation

- Ateliers vérifiés à chaque séance à partir de l'historique des dépôts.
- **Évaluation finale** : projet d'équipe sur GitHub et épreuve pratique individuelle, notés sur 20.
- Attestation de formation délivrée par OSD — Open Digital Service.

Modalités

Durée	20 heures, en 10 séances de 2 heures
Format	présentiel en salle informatique, ou intra-entreprise / établissement
Effectif	groupes de 8 à 20 participants
Matériel	un poste par participant avec Git, VS Code et un compte GitHub
Supports remis	fiches de séance, aide-mémoire des commandes Git, dépôts d'exercices
Formateur	Mohammed Slimani — développeur fullstack senior et architecte DevOps, plus de 20 ans d'expérience

Intéressé(e) par cette formation ?

Programme adaptable (GitLab, Bitbucket, volume horaire, public). Contact : contact@osdigitalis.com · WhatsApp +213 674 05 38 95 · osdigitalis.com

Open Digital Service

Mohammed Slimani — Formateur & consultant

+213 (0)6 74 05 38 95 · contact@osdigitalis.com · msslimani@osdigitalis.com

osdigitalis.com · LinkedIn : in/mohammed-slimani-osdigitalis