TABLE OF CONTENTS

QU'EST-CE QUE GIT ? 3
POURQUOI UTILISER GIT ? 4
COMMENT DÉBUTER AVEC GIT ? 4
COMMENT AMÉLIORER VOS COMMITS ? 5
COLLABORER AVEC DES MEMBRES DE L'ÉQUIPE 5
ET SI JE VEUX CONTRIBUER AU PROJET DE QUELQU'UN D'AUTRE ? 6
CELA SEMBLE ÊTRE BEAUCOUP D'ÉTAPES SUPPLÉMENTAIRES 6
CONCLUSION 6
TAGSGENERALGITTOOLSCOLLABORATION 7



GIT --- PARLONS-EN

Auteur(s) : Louis Ouellet

Pour commencer, je serai le premier à admettre que je ne suis pas un expert de Git. Je suis un débutant et j'apprends encore, mais je voulais écrire cet article pour partager ce que je sais jusqu'à présent et, j'espère, aider d'autres personnes qui commencent également. J'utilise vraiment Git de manière régulière depuis l'année dernière, principalement comme solution de sauvegarde pour mes projets. Mais j'ai réalisé que Git est bien plus qu'un simple outil de sauvegarde. C'est un atout puissant pour gérer des projets et collaborer avec d'autres.

Il y a environ un an, j'ai appris les bases des commits et pourquoi ils sont importants. Depuis, j'ai fait un effort pour améliorer mes messages de commit, en visant à les rendre aussi détaillés que possible. Récemment, j'ai commencé à explorer les Workflows et Actions de GitHub, découvrant des moyens d'automatiser certains aspects de mes projets. Alors, même si je suis encore en phase d'apprentissage, je suis ravi de partager ce que je sais avec vous. Commençons par les bases.

QU'EST-CE QUE GIT ?

Git est un système de gestion de version distribué qui vous permet de suivre les changements dans votre code. Créé par Linus Torvalds en 2005, Git est devenu un outil essentiel dans l'industrie du développement logiciel. Il aide les développeurs à gérer des projets et à travailler ensemble, en suivant chaque modification de manière fiable et structurée.

POURQUOI UTILISER GIT ?

Il existe de nombreuses raisons pour lesquelles vous devriez utiliser Git. Voici celles que je considère comme les plus importantes :

- C'est un excellent outil pour gérer vos projets.
- Il vous permet de suivre les modifications de votre code au fil du temps.
- Il facilite la collaboration avec d'autres.
- Il est étonnamment facile de commencer avec Git.

COMMENT DÉBUTER AVEC GIT ?

Pour commencer avec Git, vous devez l'installer sur votre ordinateur. Vous pouvez télécharger Git depuis le site officiel. Une fois que Git est installé, vous pouvez commencer à l'utiliser en créant un nouveau dépôt. Vous pouvez créer un nouveau dépôt en exécutant la commande suivante dans votre terminal :

```
git init
```

Cela créera un nouveau dépôt dans le répertoire actuel. Vous pouvez alors commencer à suivre les modifications de votre code en créant un nouveau commit. Vous pouvez créer un nouveau commit en exécutant la commande suivante dans votre terminal :

```
git add .
git commit -m "Votre message de commit"
```

Cela créera un nouveau commit avec toutes vos modifications. Vous pouvez ensuite pousser vos modifications vers un dépôt distant en exécutant la commande suivante dans votre terminal :

```
git push
```

Cela poussera vos modifications vers un dépôt distant. Vous pourrez alors collaborer avec d'autres en partageant votre dépôt avec eux.

COMMENT AMÉLIORER VOS COMMITS ?

Pour améliorer vos commits, vous devez suivre quelques bonnes pratiques. Voici quelques conseils pour vous aider à améliorer vos commits :

- Rédigez des messages de commit détaillés.
- Utilisez des messages de commit significatifs.
- Gardez vos commits petits et ciblés.

Un excellent outil est le mode patch. Vous pouvez utiliser le mode patch pour examiner vos modifications avant de les valider. Vous pouvez entrer en mode patch en exécutant la commande suivante dans votre terminal :

```
git add -p
```

ou

```
git add --patch
```

Cela vous permettra d'examiner vos modifications avant de les valider. Vous pourrez alors sélectionner les changements que vous souhaitez valider et ceux que vous souhaitez ignorer. De cette façon, vos commits seront plus ciblés et plus faciles à examiner.

COLLABORER AVEC DES MEMBRES DE L'ÉQUIPE

Pour collaborer avec d'autres membres de l'équipe, vous devez partager votre dépôt avec eux. Vous pouvez partager votre dépôt en les ajoutant comme collaborateurs. Vous pouvez ajouter un collaborateur en exécutant la commande suivante dans votre terminal :

```
git remote add origin git@github.com:User/UserRepo.git
```

Cela ajoutera un dépôt distant à votre dépôt local. Chaque membre de l'équipe peut maintenant créer une nouvelle branche et commencer à travailler sur ses modifications. Ils peuvent ensuite pousser leurs modifications vers le dépôt distant en exécutant la commande suivante dans leur terminal :

```
git push origin nom-de-branche
```

Lorsqu'un membre de l'équipe souhaite fusionner ses modifications avec la branche principale, il peut créer une pull request. Il peut créer une pull request en se rendant sur le site de GitHub et en cliquant sur le bouton "Nouvelle pull request". Il pourra alors examiner ses modifications et les fusionner avec la branche principale.

ET SI JE VEUX CONTRIBUER AU PROJET DE QUELQU'UN D'AUTRE ?

Si vous souhaitez contribuer au projet de quelqu'un d'autre, vous devez forker leur dépôt. Vous pouvez forker un dépôt en cliquant sur le bouton "Fork" sur le site de GitHub. Une fois que vous avez forké le dépôt, vous pouvez le cloner sur votre machine locale et commencer à travailler sur vos modifications. Vous pouvez ensuite pousser vos modifications vers votre dépôt forké et créer une pull request vers le dépôt original.

CELA SEMBLE ÊTRE BEAUCOUP D'ÉTAPES SUPPLÉMENTAIRES

Oui, cela fait pas mal d'étapes supplémentaires, mais elles valent la peine pour le contrôle et la clarté que Git fournit. De plus, GitHub propose des outils comme les Workflows et les Actions pour automatiser certains de ces processus. Par exemple, vous pouvez créer un workflow qui exécute des tests et construit votre projet à chaque push de modifications. Vous pouvez même automatiser la fusion des pull requests après approbation, ce qui vous fait gagner du temps et rend vos projets plus efficaces.

CONCLUSION

Git est un outil puissant qui peut vous aider à gérer vos projets et à collaborer avec d'autres. Il est facile de commencer avec Git et de nombreuses ressources sont disponibles pour vous aider

à apprendre. J’espère que cet article vous a aidé à comprendre les bases de Git et comment vous pouvez l’utiliser pour améliorer vos projets. Si vous avez des questions ou besoin d’aide, n'hésitez pas à demander. Je suis toujours heureux d’aider. Bon codage !

TAGSGENERALGITTOOLSCOLLABORATION

- Twitter
- Facebook
- LinkedIn
- Reddit
- Telegram
- Email

View the discussion thread.

From:
https://laswitchtech.com/ - **LaswitchTech**

Permanent link:
https://laswitchtech.com/fr/blog/2024/11/07/git-let-s-talk-about-it

Last update: **2026/02/16 13:45**