

git-cmds-branchs-depot-unique

Alice et Bob collaborent à l'aide de branches	1
le problème	1
Dépôt initial et invitations	1
Bob clone le projet d'Alice	1
Le travail collaboratif	1
Bob se met à jour	3
Commandes utiles sur les branches	3
Renommage de branche	3
Modification d'une branche distante	4
Suppression d'une branche distante	4

Alice et Bob collaborent à l'aide de branches

le problème

Supposons qu'Alice et Bob veulent collaborer sur le projet super-proj

- Alice possède le code initial qu'elle publie sur son dépôt gitlab : <https://gitlab.com/alice/super-proj.git>

Dépôt initial et invitations

Alice invite Bob sur son projet en tant que developer (droit de lecture et écriture) ou maintenir (tous les droits).

Bob clone le projet d'Alice

Bob va faire un clone du dépôt d'Alice sur Gitlab, pour obtenir une copie locale :

```
git clone https://gitlab.com/alice/super-proj.git
```

- chacun vérifie qu'il a un remote : origin en tapant :

```
git remote -v
```

(qui doit afficher 2 lignes)

Le travail collaboratif

Bob crée une branche et l'utilise :

```
git branch bd  
git checkout bd
```

ou en une seule ligne :

```
git checkout -b bd
```

Bob fait le travail/Teste puis commite :

```
git commit -am "Intégration BD"
```

Bob pousse sa branche sur son remote origin :

```
git push origin bd
```

Bob fait un merge request :

Bob se rend sur le dépôt d'Alice et y fait un Merge Request ou MR (Gitlab) ou Pull Request ou PR (sur Github ou Bitbucket)

Alice accepte directement le Merge Request :

Alice peut accepter automatiquement le Merge Request de Bob si celui-ci est très simple, directement sur le Gitlab. Dans ce cas elle fait un simple pull (git pull) de son main pour que sa copie locale soit à jour aussi.

Alice teste le code de Bob Alice récupère la branche de Bob :

```
git fetch bob bd
```

ou

```
git fetch bob bd --tags
```

pour aussi récupérer des tags éventuels.

(ou télécharge directement la branche concernée depuis Gitlab)

Alice consulte la branche de Bob :

```
git checkout Bob/bd
```

(pour lister toutes les branches : `git branch -av`)

Alice revient dans son main :

```
git checkout mainer
```

Alice traite le Merge Request : Alice se rend sur son dépôt Gitlab, puis refuse ou accepte le Merge Request. Si elle l'accepte, met à jour sa copie locale :

```
git pull
```

Bob se met à jour

Il reste à Bob à récupérer le main mis à jour par Alice pour se mettre aussi à jour.

Bob se met à jour avec un fetch et un merge :

```
git fetch origin main  
git merge origin/main
```

Bob se met à jour directement avec un pull :

```
git pull origin main
```

Commandes utiles sur les branches

Renommage de branche

```
git branch -m <oldname> <newname>
```

Modification d'une branche distante

Si vous voulez mettre à jour une branche distante mabranche :

```
git push origin mabranche --force
```

A utiliser si d'autres n'ont pas déjà intégré votre branche par exemple ...

Suppression d'une branche distante

```
git push origin --delete mabranche
```

2022 Gérard Rozsavolgyi roza [a] univ-orleans.fr