

git-cmds-branches

Alice et Bob collaborent à l'aide de branches	1
le problème	1
Le travail collaboratif	2
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>
- Bob fait un fork du projet d'Alice sur Gitlab
- il obtient ainsi sa propre copie de super-proj à l'URL : <https://gitlab.com/bob/super-proj.git>, qu'il peut cloner pour en obtenir une copie locale :

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

- Alice invite Bob sur son projet en tant que Reporter
- Bob invite Alice sur son projet en tant que Reporter
- Alice ajoute le remote de Bob :

```
git remote add bob https://gitlab.com/bob/super-proj.git
```

- Bob ajoute le remote d'Alice :

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

- chacun vérifie qu'il a 2 remotes : origin et celui de son collaborateur en tapant :

```
git remote -v
```

(qui doit afficher 4 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 :

```
git push origin bd
```

Bob fait un merge request :

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

Alice accepte le Merge Request :

Alice peut accepter directement 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 d'abord la branche de Bob :

```
git fetch bob bd
```

ou

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

pour aussi récupérer des tags éventuels.

Alice consulte ensuite la branche de Bob :

```
git checkout Bob/bd
```

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

Alice revient dans son main :

```
git checkout main
```

Alice accepte le Merge Request :

Alice se rend sur son dépôt Gitlab, puis accepte le Merge Request et met à jour sa copie locale :

```
git pull
```

Bob se met à jour

Il reste à Bob à récupérer le main d'Alice pour se mettre à jour

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

```
git fetch alice main  
git merge alice/main
```

ou Bob se met à jour directement avec un pull :

```
git pull alice 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 ...

Suppression d'une branche distante

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

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