

Riassunto delle due modalità che abbiamo visto per gestire i progetti in team con git e gitlab.

Ce ne possono essere anche varie altre, dipende dalle scelte aziendali.

Lo scopo è per tutte quello di fare il merge delle modifiche dei diversi sviluppatori solo quando si è confidenti che esse siano corrette, non provochino “regressione” e non vadano in contrasto con modifiche fatte da altri.

Primo metodo visto – lavorare sul branch

- Da locale -> Si fa il clone del repository remoto in locale
- Da locale -> Si crea un branch per le modifiche da effettuare
- Da locale -> Si fanno le modifiche, si aggiungono e si fa commit del branch in locale
- Da locale -> Ci si sincronizza con il repository remoto. Questo vuol dire valutare in locale (prima di andare sul repository remoto) se qualcun altro ha fatto modifiche nel frattempo al progetto. Si può fare mediante un pull del main remoto sul proprio branch o usare il comando rebase.
- Da locale -> Si fa il push del branch sul repository remoto
- Da remoto -> Si fa una merge request del branch remoto sul main remoto
- Dopo che il merge è stato valutato ed accettato da remoto dal Leader di progetto, per completezza si dovrebbero poi eliminare i branch locali e remoti prima di continuare a lavorare

Secondo metodo visto – lavorare sul fork

- Da remoto -> Si fa il fork del repository sul proprio account personale
- Da locale -> Si fa il clone del repository remoto creato con il fork, in locale
- Da locale -> Si fanno le modifiche, si aggiungono e si fa commit del repository in locale
- Da locale -> si aggiunge il repository originale come remoto con il comando (git remote add upstream <https://gitlab.com/original-owner/nome-repository.git>). Se non si fa, il repository ottenuto dal fork non sarà allineabile con le modifiche fatte da altri al repository originale
- Da locale -> si aggiorna il fork con il repository originale mediante il pull. Questo vuol dire valutare in locale (prima di andare sul repository remoto) se qualcun altro ha fatto modifiche nel frattempo al progetto
- Da locale -> si fa il push delle modifiche nel fork su remoto
- Da remoto -> Si fa una merge request del branch remoto sul main remoto
- Dopo che il merge è stato valutato ed accettato da remoto dal Leader di progetto, occorre sincronizzare nuovamente il repository ottenuto dal fork in locale, prima di continuare a lavorare.

Nota: le azioni in blu non le abbiamo viste durante le esercitazioni per semplificare il processo.