

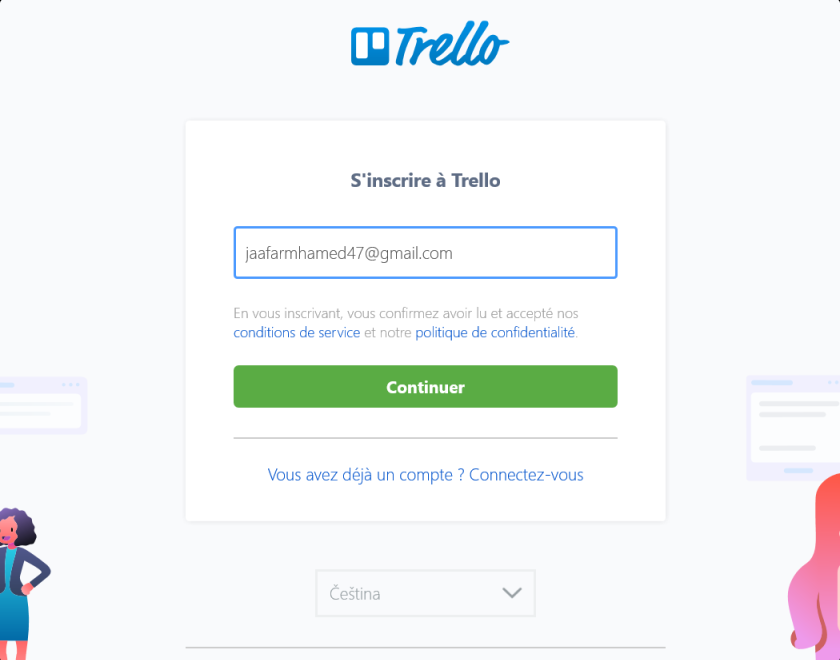
***rapport de brief pROJET***

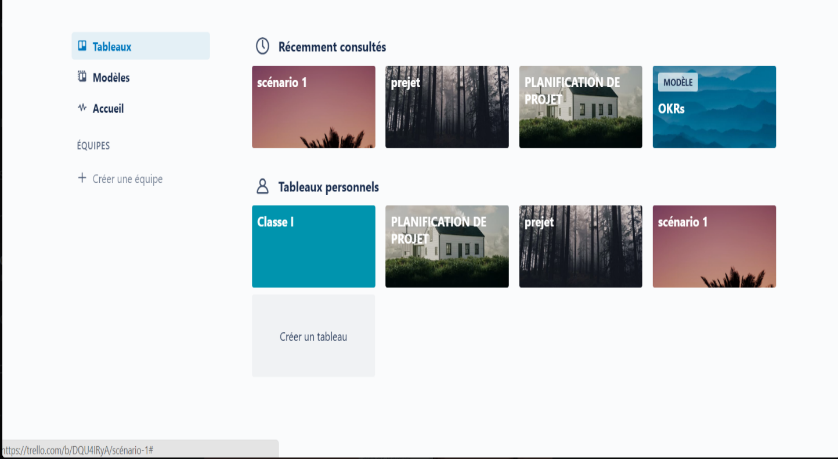
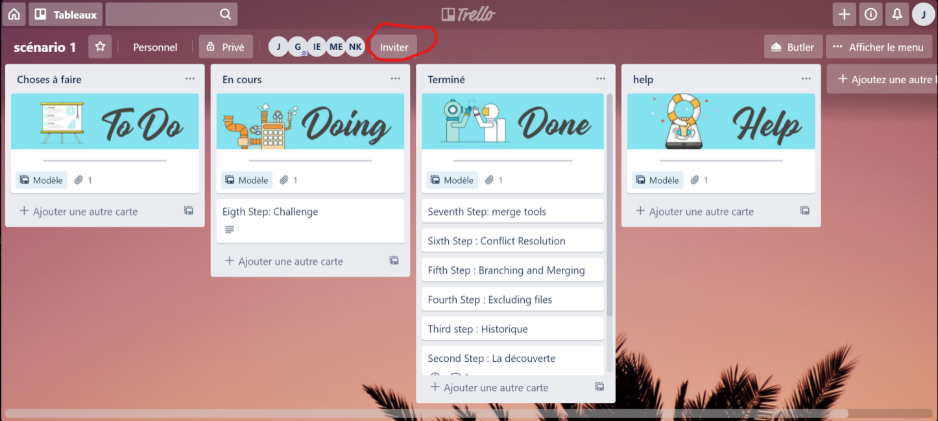
***SCEnario 1***



29 novembre 2019

**Création du tableau Trello pour la planification du Projet**

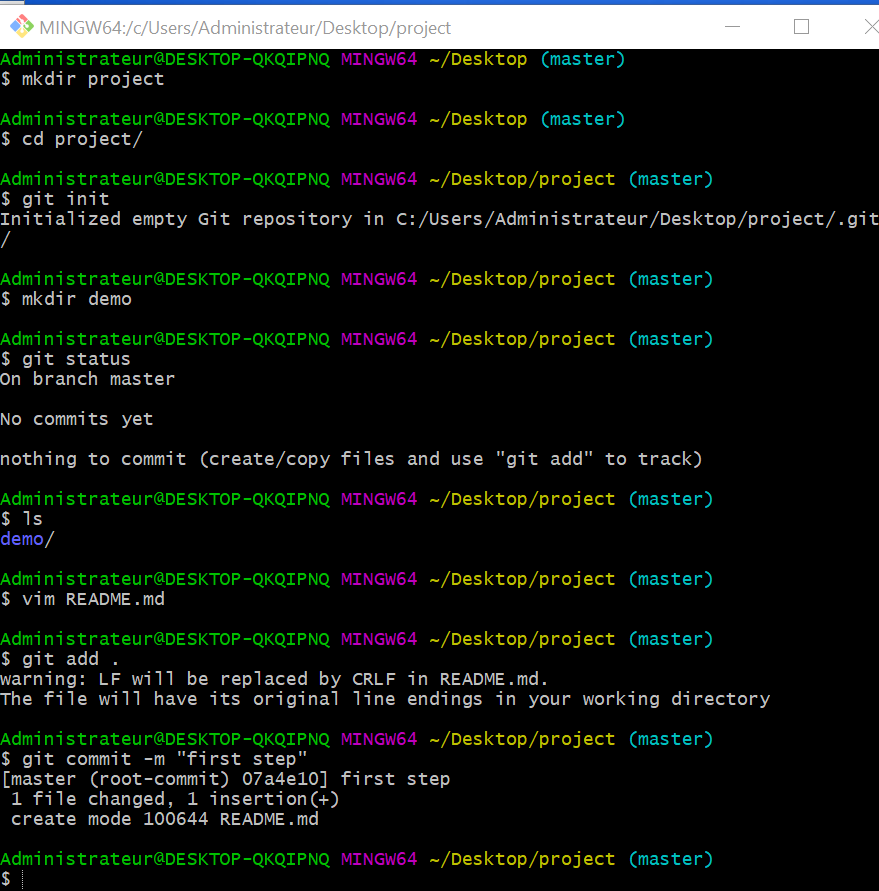
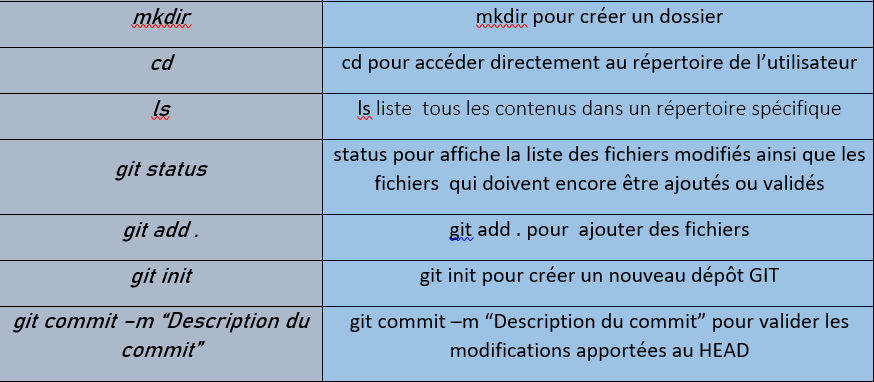
****



First Step : Immersion

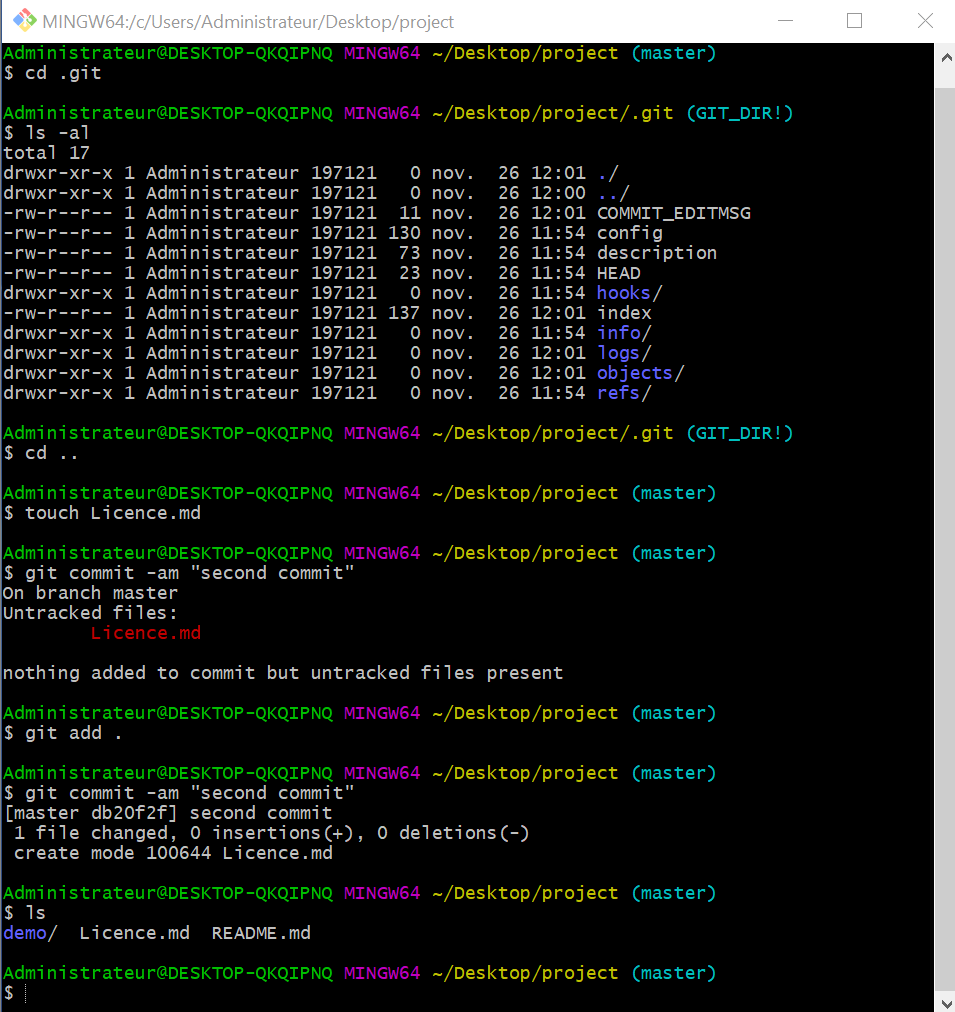
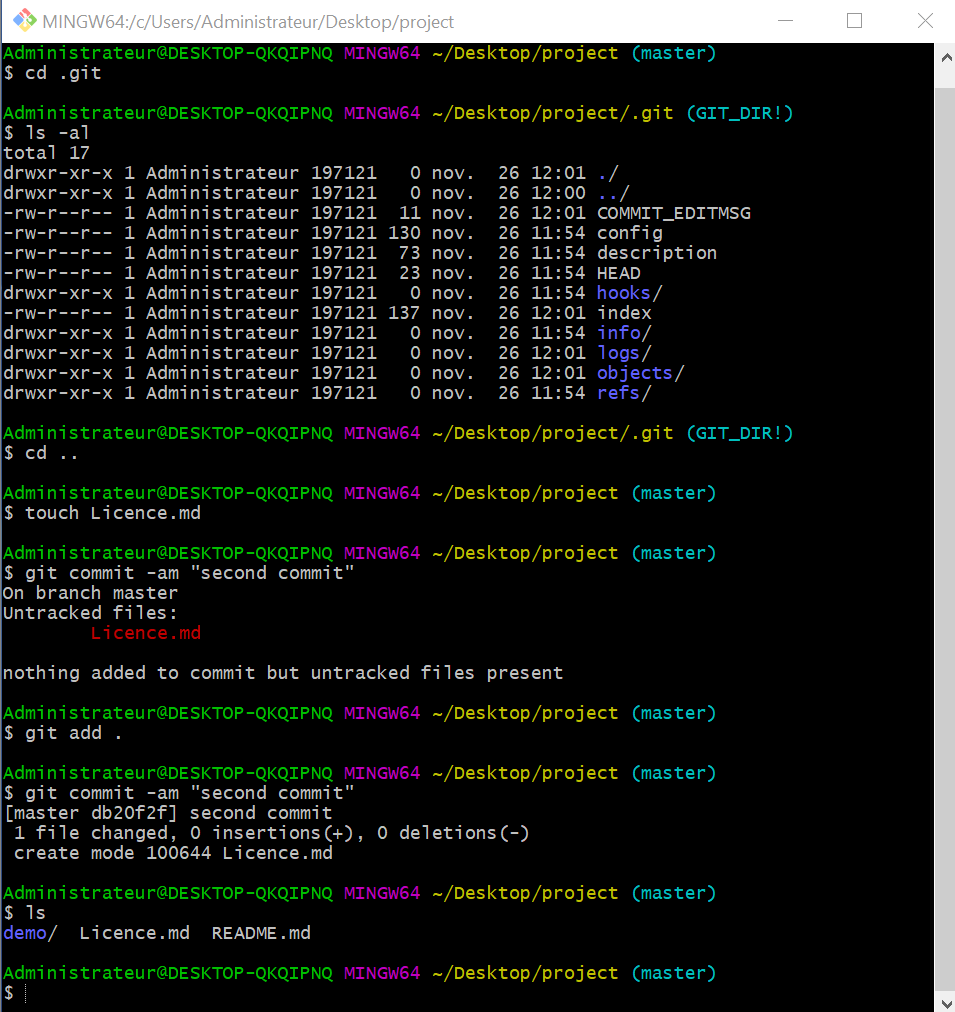
Git s’est imposé comme le gestionnaire de versions de référence. Il est utilisé par de nombreux projets Open Source (hébergés par GitHub) , la plupart des sociétés l’ont désormais adopté en remplacement des ancêtres CVS et SVN.

**Explication de fonction «. git » :** c’est un dossier temporaire pour initialiser le dépôt local.

 **Explication des commandes :**

La commande Vim READM.me : pour ajouter la line « Demo project un simple fichier »

Second Step: La découverte

****

**HEAD :** The HEAD in Git is the pointer to the current branch reference, which is in turn a pointer to the last commit you made or the last commit that was checked out into your working directory. That also means it will be the parent of the next commit you do.

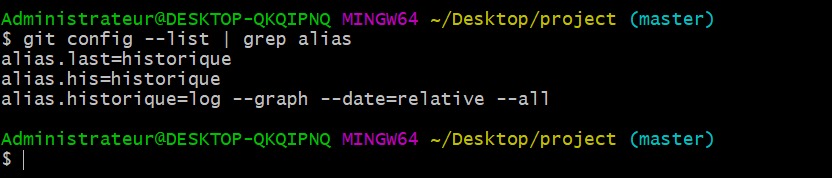
**LOGS :** une clause pour afficher l'historique des commit traiter.

**Une branche :** dans git est tout simplement un pointeur vers un de ces objets commit.la branche par défaut dans git appeler master il pointe vers le dernier commit réalisés.

**La *commande* :** Ls –al affiche les fichiers cachés.

Third step : Historique





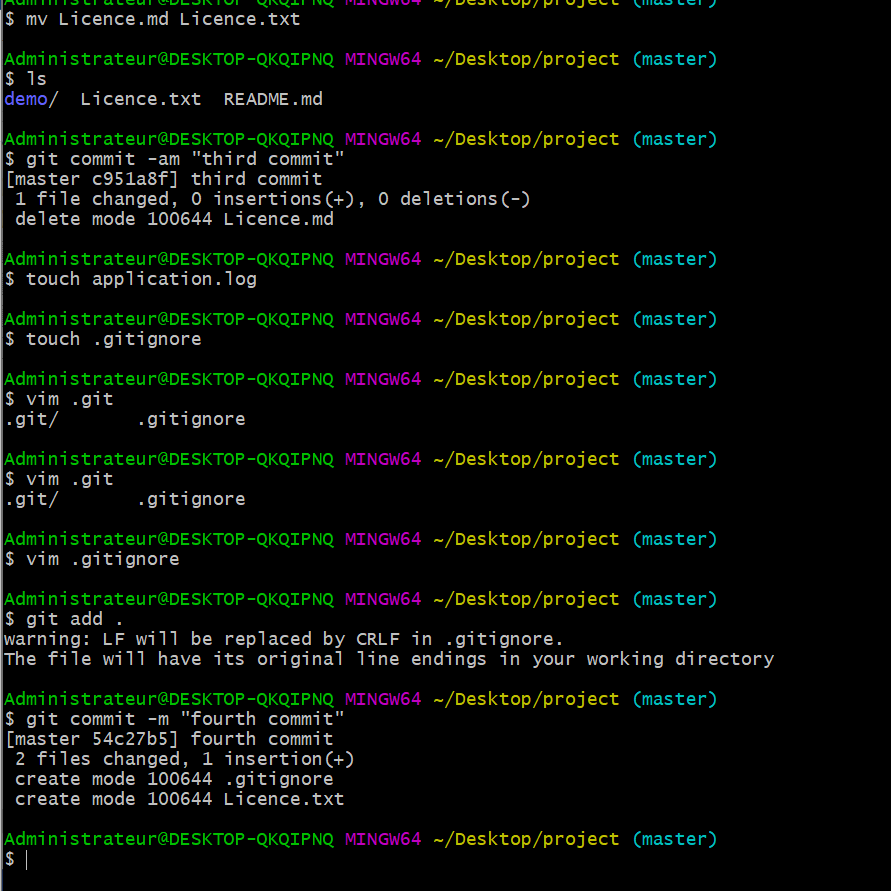
**LES COMMANDES :**

**Git log --oneline** : afficher le dernier commit .

**Git historique :** pour afficher les historique du fichier README .md avec alias.

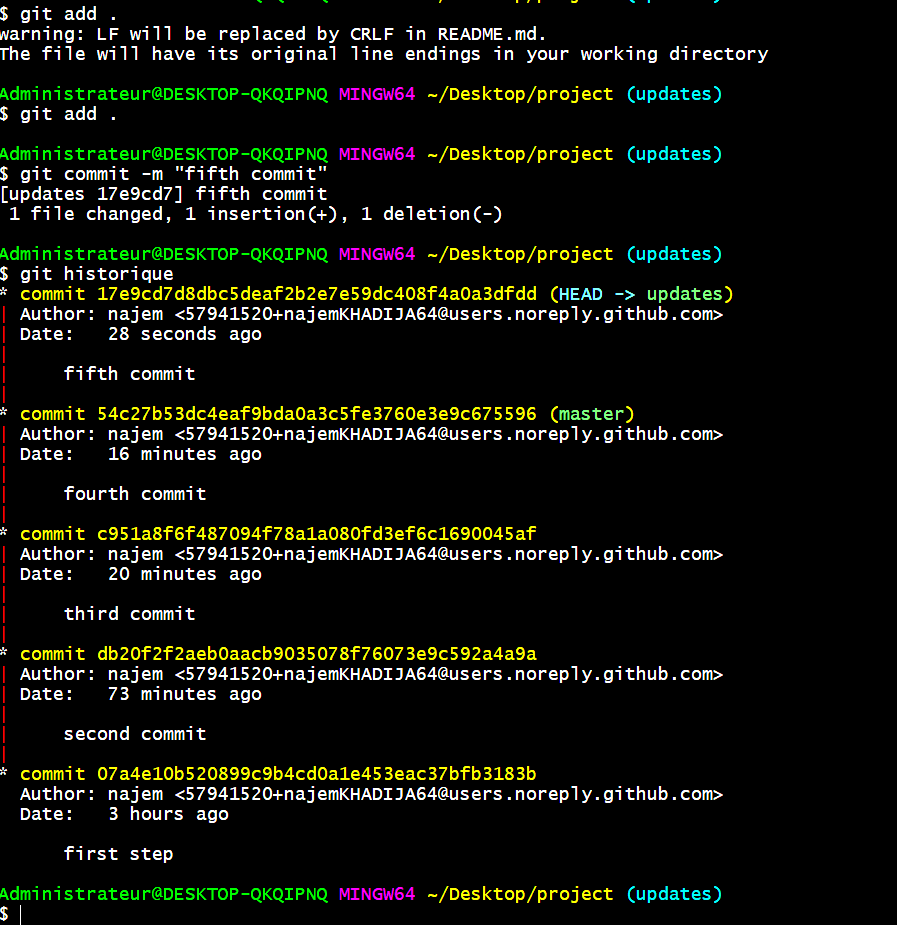
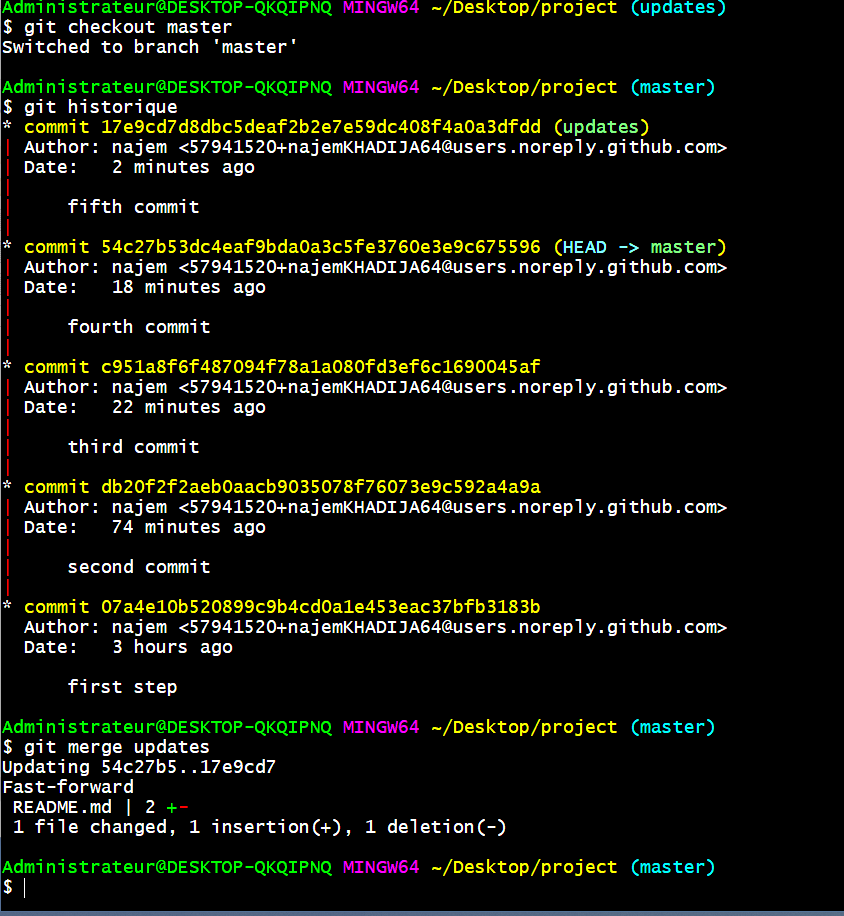
Afficher la liste des alias .

Fourth Step : Excluding files



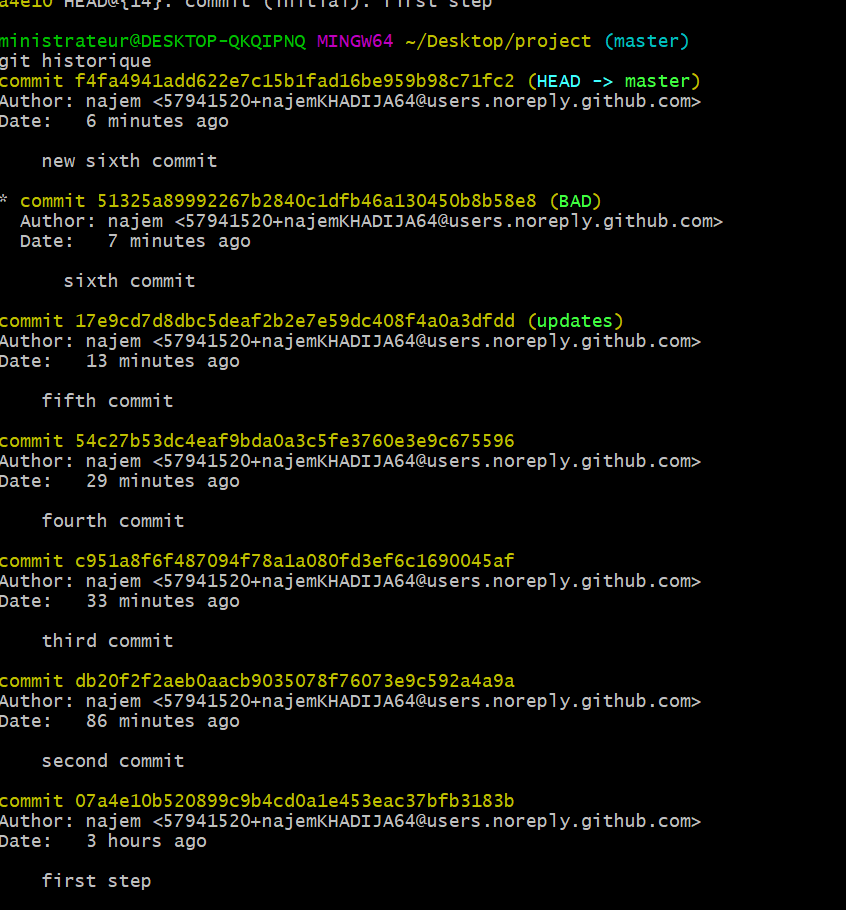
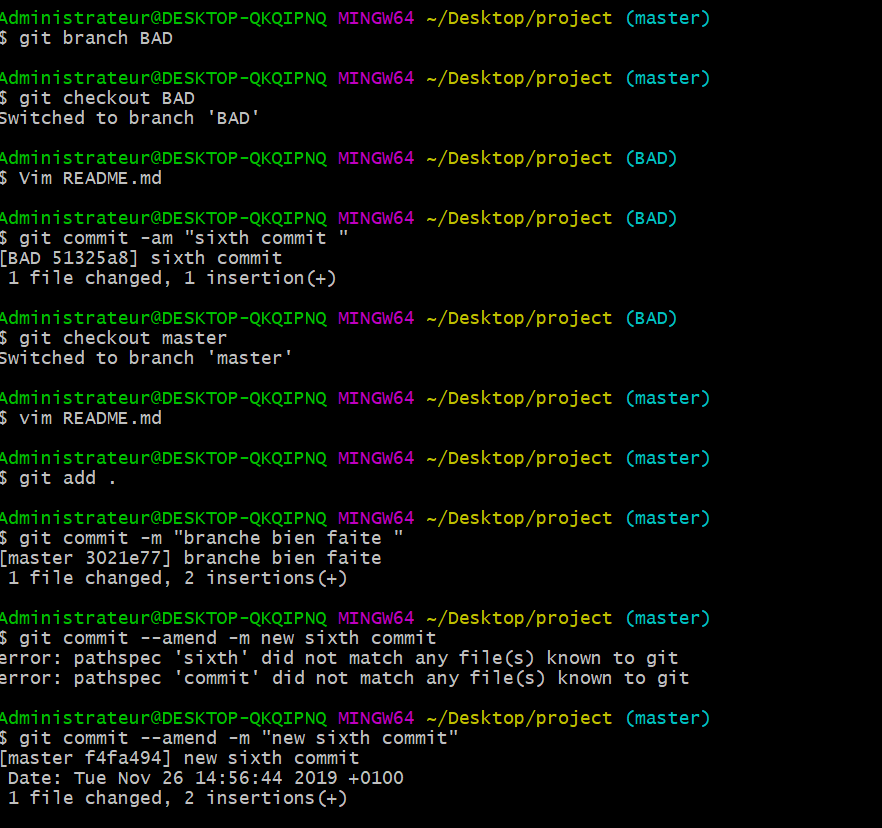
Le fichier nommé “.gitignore” ignore tous les fichiers non trackés.

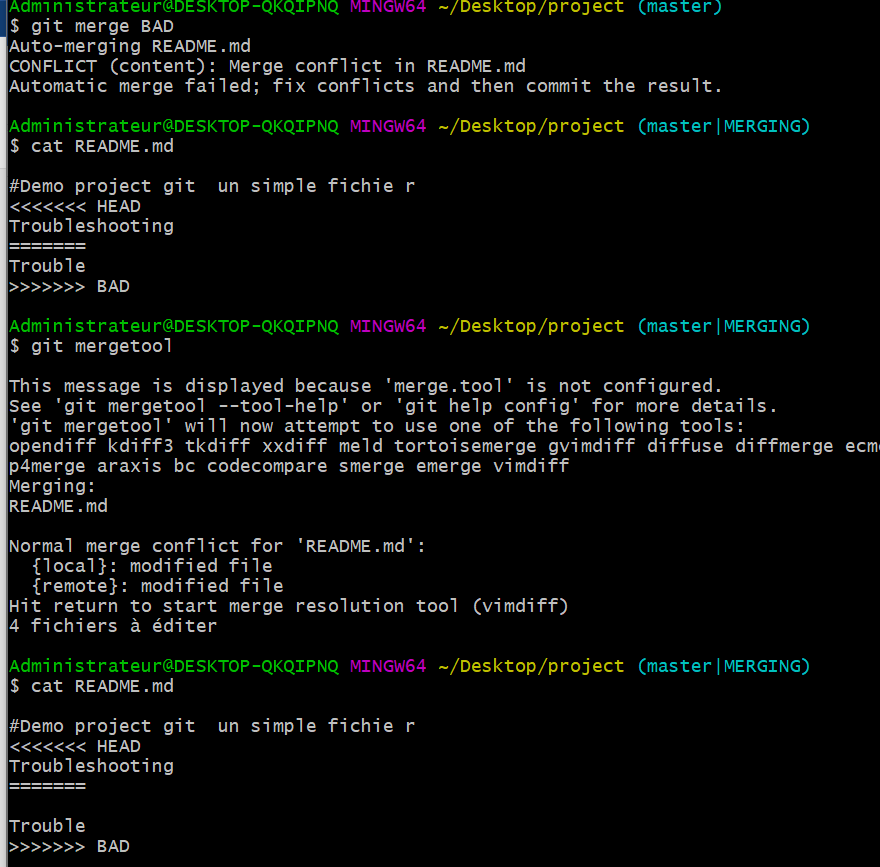
**Fifth Step: Branching and Merging**

****

* Après la création de la branche Updates , on constate que le nom de la branche s'affichera
* when we add Updates branch , the branch master appears while tapping alias command. But before it didn't appear.

Sixth Step : Conflict

******

******

Après la modification du commentaire ‘branche bien faite’ ======> ceci bien modifier en ' new sixth commit ***.***

On a un conflit à cause de la modification du même fichier ' [README.md](http://readme.md/)' dans des branches différentes ( master et BAD ).

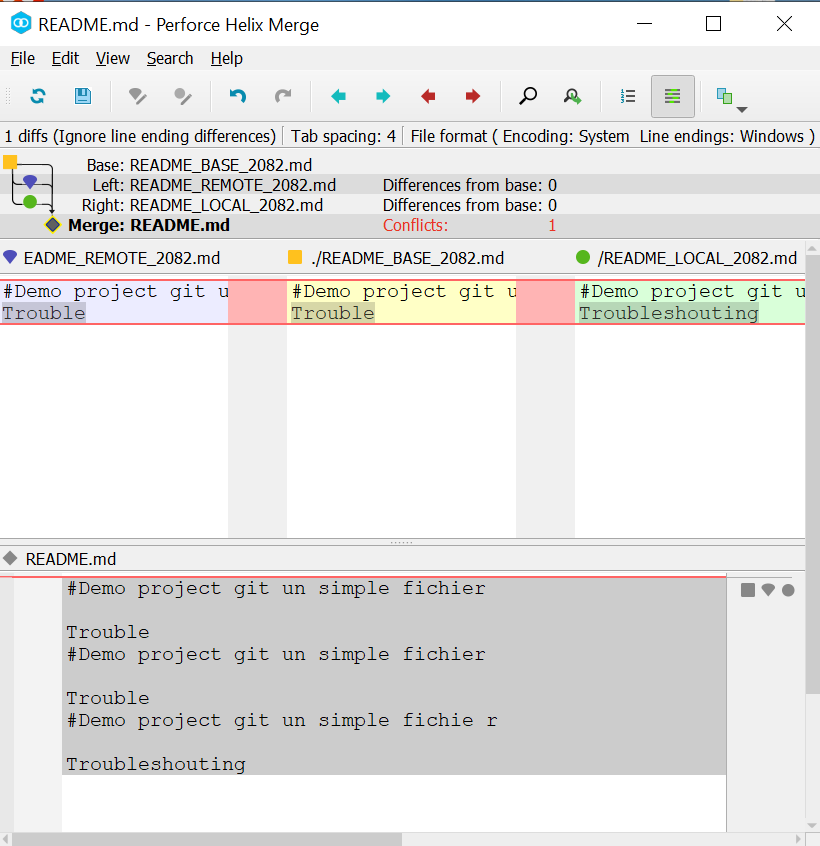
After merging branches( master and BAD ) the content of both of them was showed inside [README.md](http://readme.md/) file with a name of each branches.

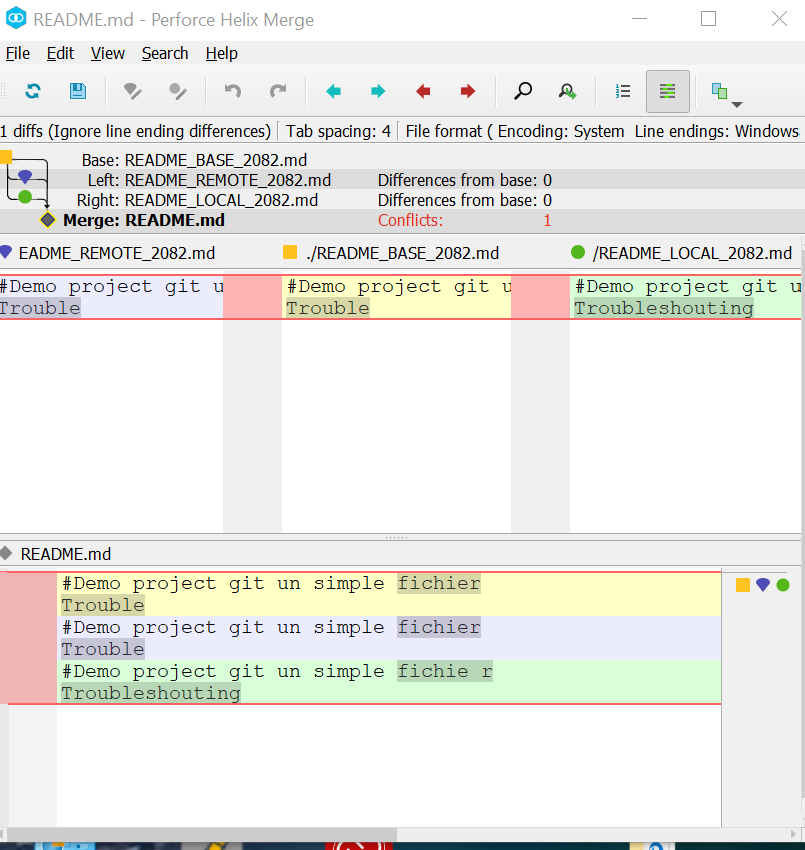
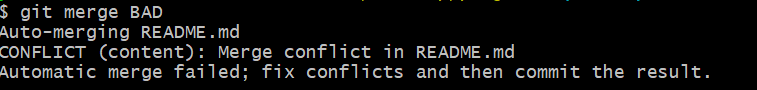
On a besoin de la configuration de merge.tool pour faire le merge avec succès.

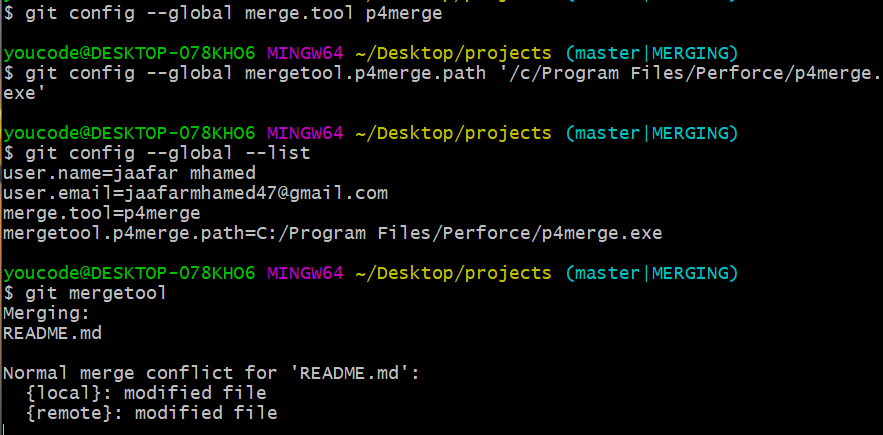
**Seventh Step: merge tools**

Installation de p4merge :

<https://www.perforce.com/products/helix-core-apps/merge-diff-tool-p4merge>

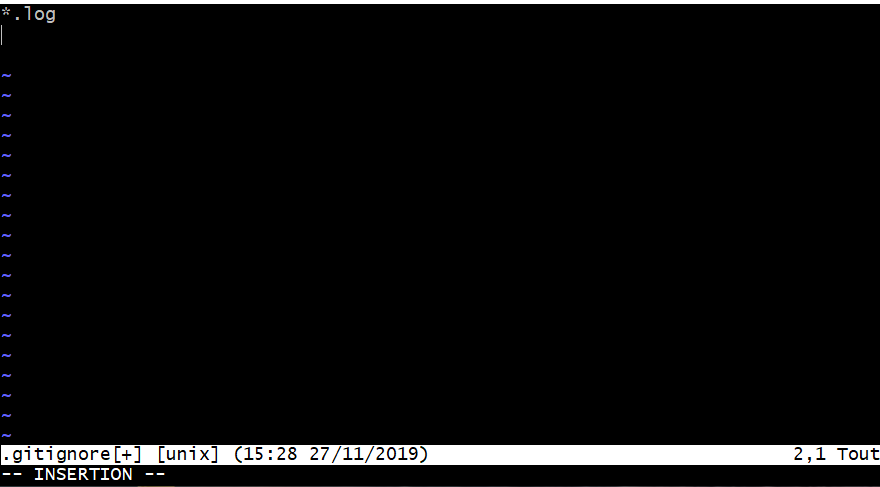


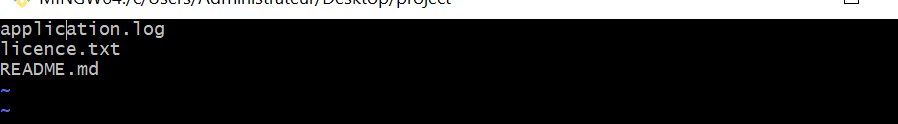




La commande Git config –global mergetool.p4merge.path “lien d’installation” ( .exe) Nous permettre pour faire merge tool.En fin on a constate que la branche BAD et master est merge .

Eigth Step: Challenge





* on a ajouté les fichiers **licence.txt Readme.md .gitignore** dans gitignore .