



# ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS

INNOVACIÓN PARA LA EXCELENCIA

Comandos básicos de Git

Laines Francisco

1946

Osorio Cesar

Esta es una recopilación de los códigos más importantes y necesarios para un uso comun de la herramienta Git.

#### 1. Configuración básica

- *Nombre que aparecerá en los commits:*  
Git config —global user.name “Nombre”
- *Configurar email:*  
Git config —global user.email email

#### 2. Iniciando repositorio

- *Iniciando la carpeta donde se encuentra el proyecto:*  
Git init
- *Clonar el repositorio de github o bitbucket:*  
Git clone <dirección de la carpeta>
- *Hacer el primer commit:*  
Git commit -m “identificación”
- *Subir el repositorio:*  
Git push origin master

#### 3. Git Add

- *Añadir todos los archivos para el commit:*  
Git add .
- *Añadimos el archivo para el commit:*  
Git add <archivo>
- *Añadimos todos los archivos para el commit omitiendo los nuevos:*  
Git add --all
- *Añadimos todos los archivos con la extensión especificada:*  
Git add \*.extensión

#### 4. Git commit

- *Cargar en el HEAD los cambios realizados:*  
Git commit -m “Identificación”
- *Agregar y Cargar en el HEAD los cambios realizados:*

- *De haber conflictos los muestra:*  
Git commit -a
- *Agregar al último commit, este no se muestra como un nuevo commit en los logs:*  
Git commit --amend -m “Etiqueta”

#### 5. Git Log

- *Lista de commits:*  
Git log
- *Muestra los cambios en los commits:*  
Git log --oneline --stat
- *Muestra los gráficos de los commits:*  
Git log --oneline --graph

#### 6. Git diff

- *Muestra los cambios realizados a un archivo:*  
Git diff  
Git diff --staged

#### 7. Git head

- *Sacar un archivo del commit:*  
Git reset HEAD <archivo>
- *Devuelve el ultimo commit que se hizo y pone los cambios en staging:*  
Git reset --soft HEAD^
- *Devuelve el ultimo commit y todos los cambios:*  
Git reset --hard HEAD^
- *Devuelve los dos ultimos commits y todos los cambios:*  
Git reset --hard HEAD^^
- *Rollback merge/commit:*  
Git log  
Git reset --hard <commit\_sha>

Git commit -a -m "Etiqueta"

8. *Git Branch*

- *Crea un branch:*  
*Git branch <nameBranch>*
- *Lista de branches:*  
*Git branch*
- *Comando -d elimina el branch y lo une al master:*  
*Git branch -d <nameBranch>*
- *Elimina sin preguntar*  
*Git branch -D <nameBranch>*

9. *Otros comandos*

- *Lista un estado actual del repositorio con lista de archivos modificados o agregados:*  
*Git status*
- *Quita del HEAD un archivo y le pone el estado de no trabajado:*  
*Git cheackout -- <file>*

*Head: último commit de la rama en la que se esté trabajando.*