



# GIT - Control de Versiones de Código

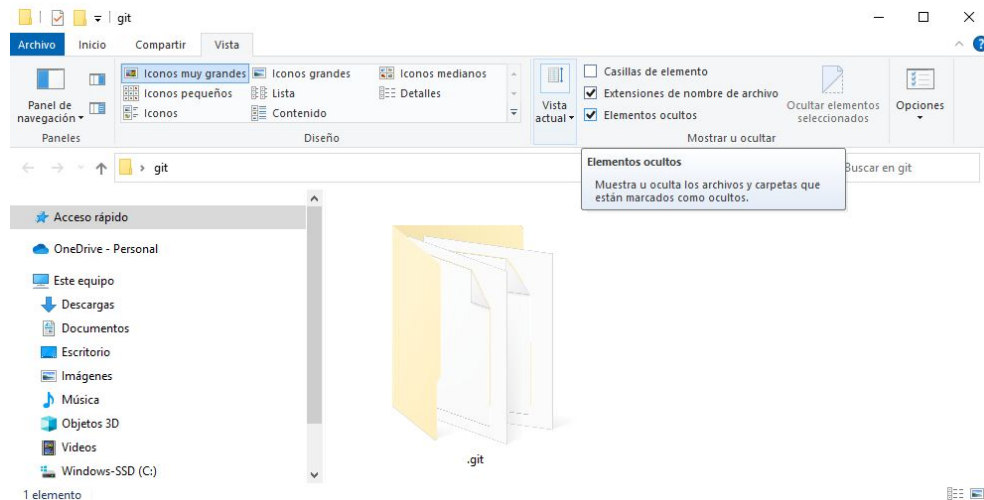
## LA HERRAMIENTA GIT Y SU USO

## LA HERRAMIENTA GIT Y SU USO

- Crear un nuevo repositorio GIT.
- Entendiendo directorios GIT y primeros comandos de respaldo.
- Manejo de ramas en GIT.

- Crear un nuevo repositorio GIT

**git init** (Inicia un nuevo repositorio)



**Utilizando este comando en consola nos creará un nuevo proyecto Git, el mismo estará en una carpeta oculta y con él podremos interactuar a través de la consola.**

- Entendiendo directorios GIT y primeros comandos de respaldo.

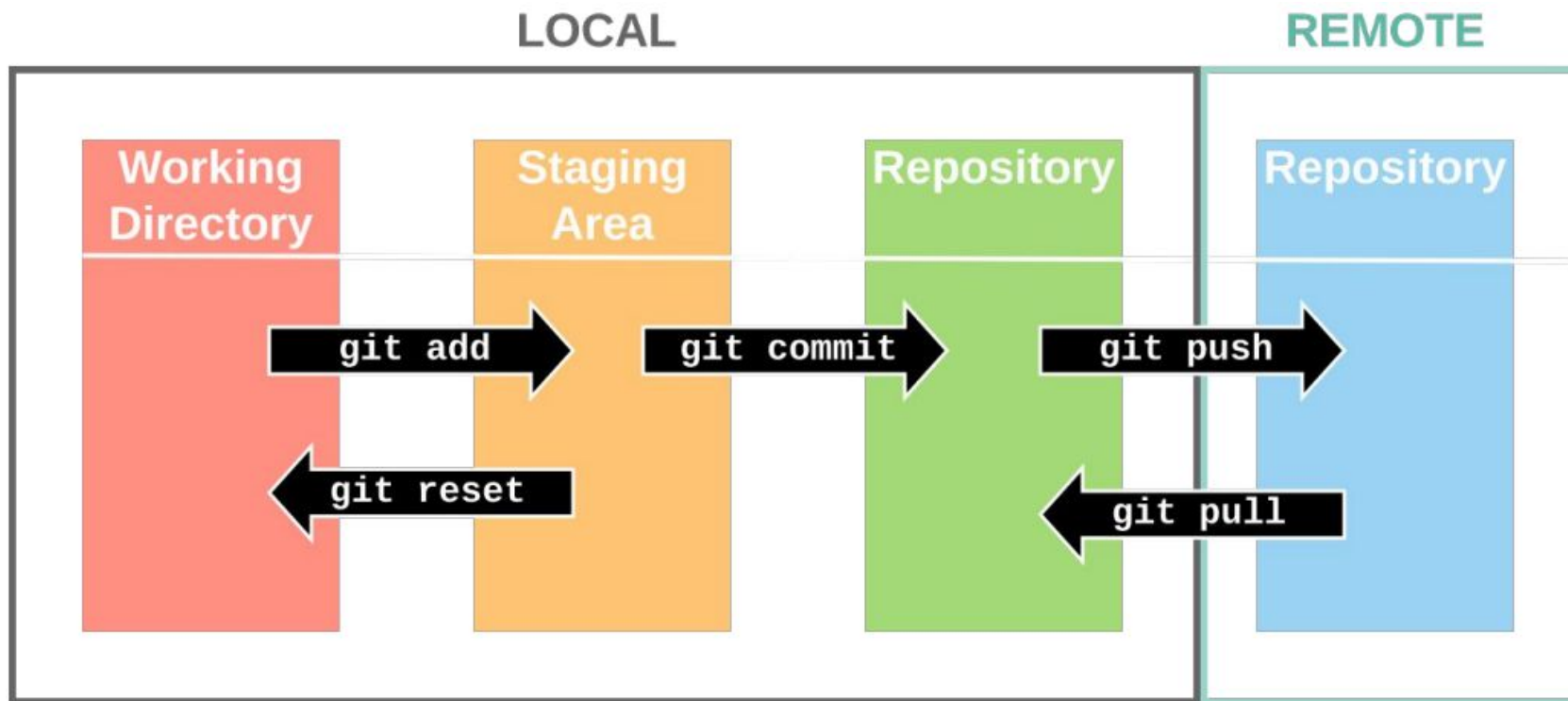
# Áreas de trabajo en GIT



- Entendiendo directorios GIT y primeros comandos de respaldo.

En Git manejamos áreas de trabajo, la del directorio local "Working Directory", la de los archivos que nos interesa respaldar "Staging Area ", la de los archivos respaldados "Repository"

- Entendiendo directorios GIT y primeros comandos de respaldo.



- Entendiendo directorios GIT y primeros comandos de respaldo.

# Comandos básicos de GIT



- Entendiendo directorios GIT y primeros comandos de respaldo.

**git status** (Muestra la lista de los archivos que se han cambiado junto con los archivos que están por ser preparados o confirmados.)

**git add .** (Añade cambios del directorio de trabajo en el “staging area”)

**git commit -m “comentario”** (captura una instantánea de los cambios preparados en ese momento del proyecto.)

**git log** (Lista los commit realizados)

**git reset .** (Borra cambios del directorio de trabajo en el “staging area”)



- Manejo de ramas en GIT.

**git branch** (Muestra ramas existentes.)

**git branch nueva-rama** (Crea una nueva rama.)

**git checkout nombre-rama** (Cambia a la rama nombrada)

**git checkout -- .** (Vuelve todos los archivos al último commit)

**git checkout -d nombre-rama** (Borra rama seleccionada)

**git merge nombre-rama** (Fusiona rama actual con la nombrada)

