

Introducción a Git y GitHub

DEV.F
DESARROLLAMOS(PERSONAS);

dev

Sistema de control de versiones (VCS)

Un **sistema de control de versiones (VCS - Version control system)** es aquel que nos permite llevar un historial y control de cambios a medida que las personas y los equipos colaboran en proyectos juntos.

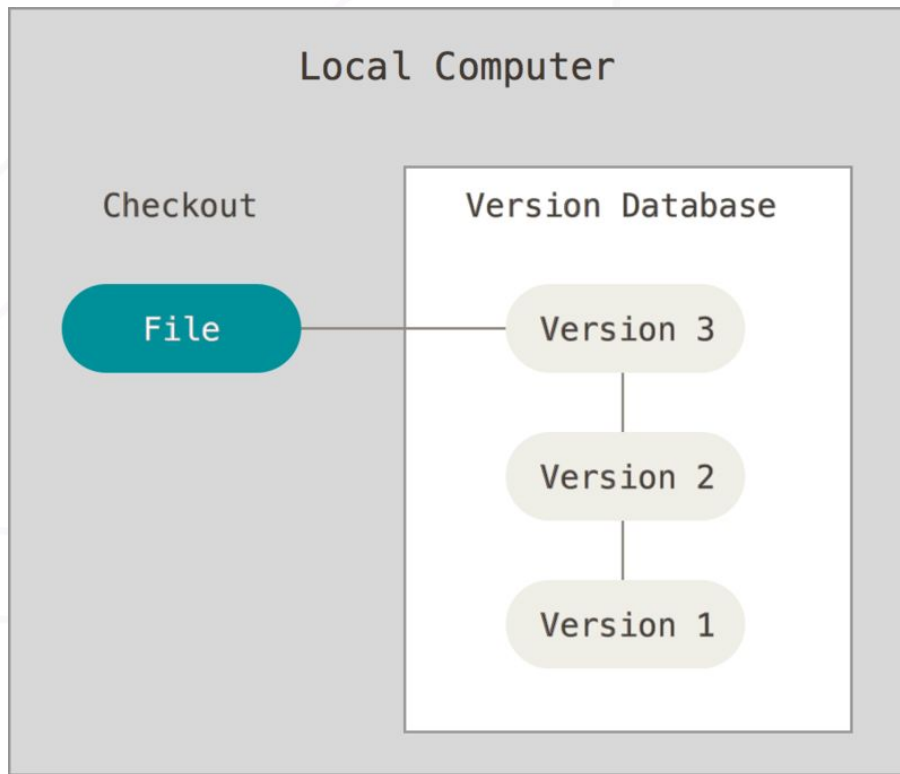
¿Que cambios se hicieron?

¿Quién hizo los cambios?

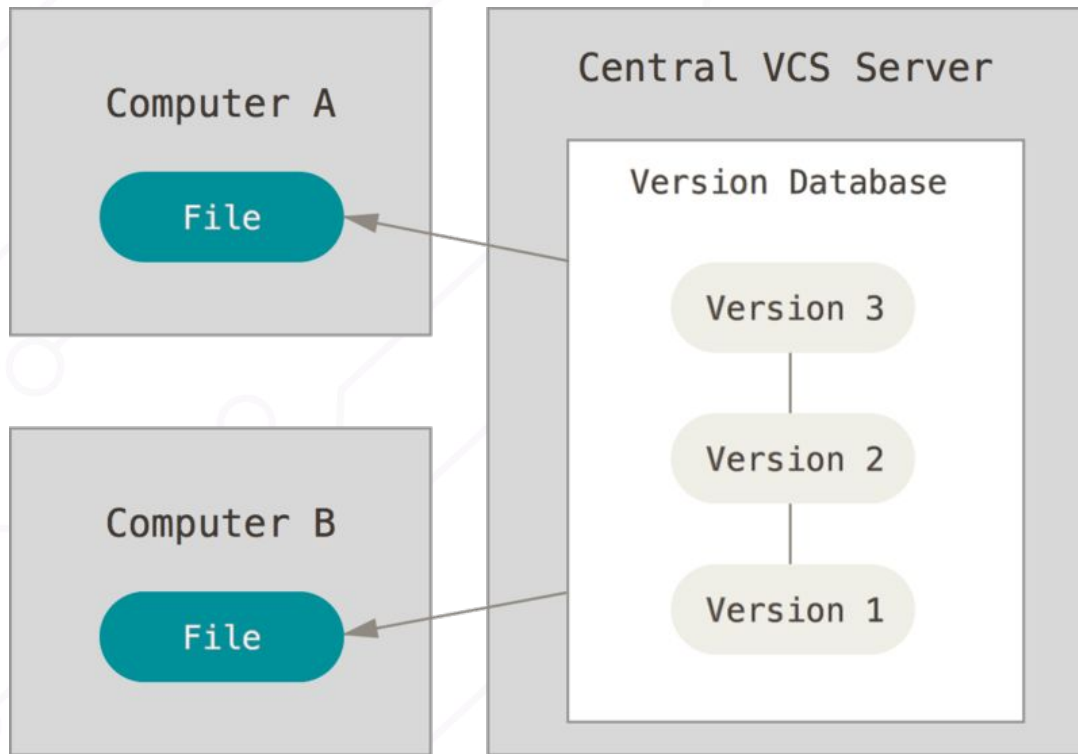
¿Cuando se hicieron los cambios?

¿Por qué fueron requeridos los cambios?

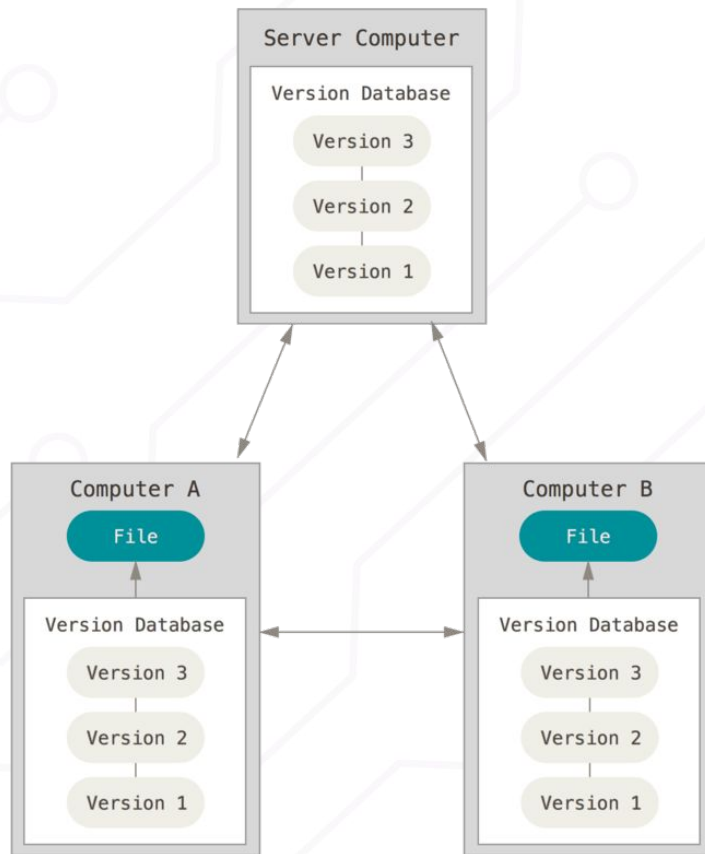
Tipos de VCS (Local)



Tipos de VCS (Centralizado)



Tipos de VCS (Distribuido)





Git es un (VCS) de tipo **distribuido** de código abierto y actualmente el más usado por los desarrolladores gracias a sus beneficios para individuos y equipos de trabajo.

- Acceso detallado a la historia del proyecto
- Colaboración en cualquier momento y lugar
- Entre otros...

Su uso principal es mediante Interfaz de línea de comandos (**CLI - Command line interface**)

¿Qué es un repositorio?

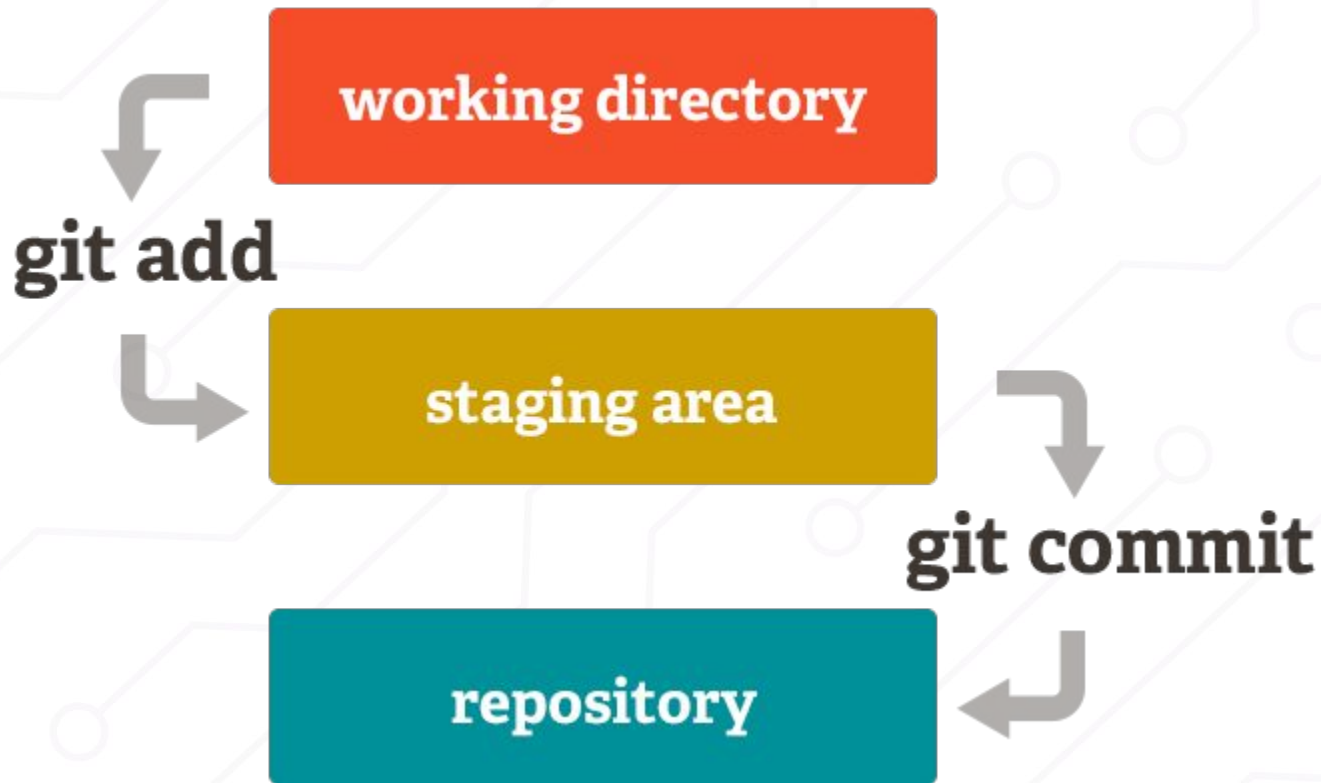


Un **repositorio** es un espacio de almacenamiento donde se organiza, mantiene y difunde información.

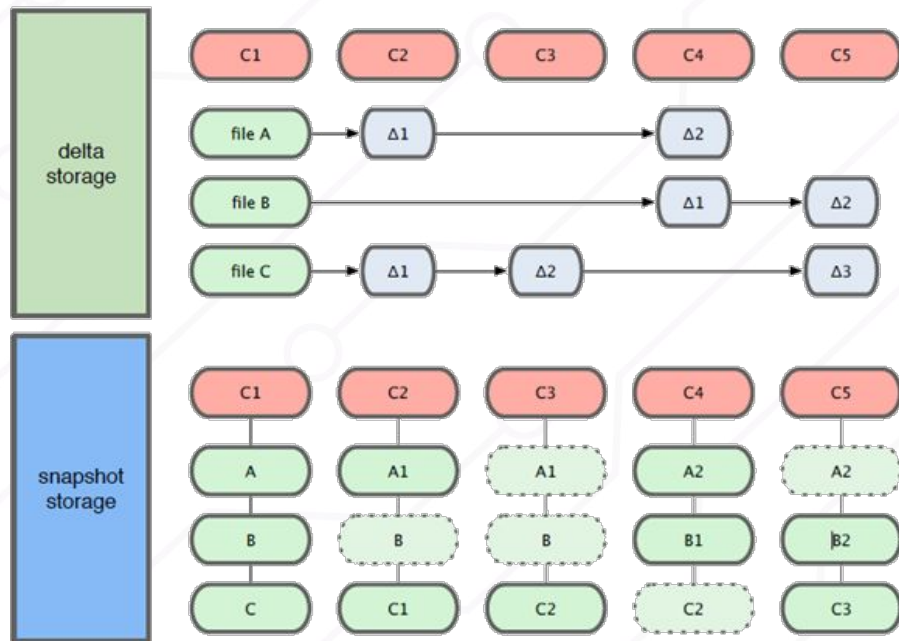
En el caso de **Git** el **repositorio** viene siendo el proyecto donde estará la colección de archivos y carpetas junto al historial de cambios.



Estados de Git



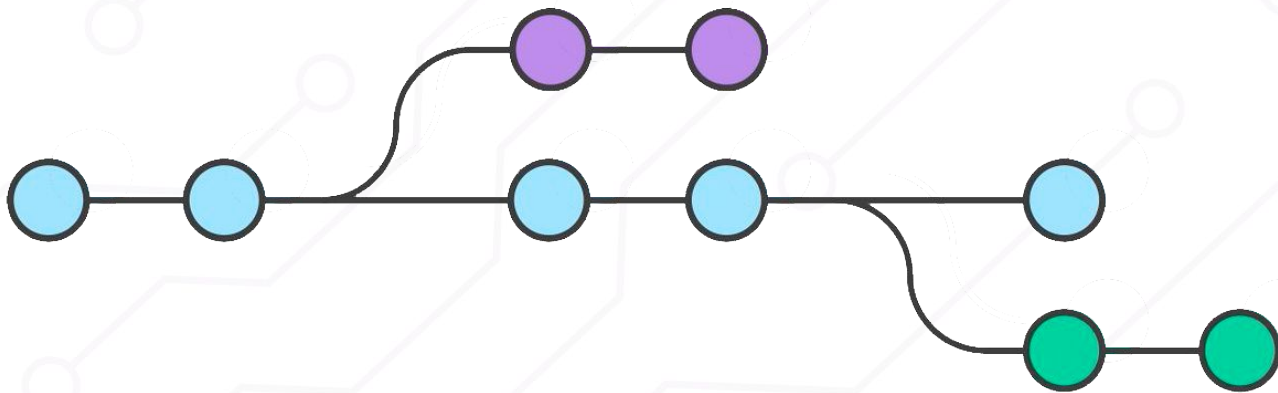
Instantáneas (Snapshots)



Un **Snapshot** es un estado de 'algo' en el tiempo, **Git** maneja **Snapshots** no cambios.

Ramas (Branch)

Las **ramas (Branch)** son bifurcaciones o variantes de un repositorio, estas pueden contener diferentes archivos y carpetas o tener todo igual excepto por algunas líneas de código.



Instalación y manejo de Git

DEV.F
DESARROLLAMOS(PERSONAS);

dev



git



DEV.F



Configuración de Git

- Desde consola, se puede acceder a la configuración de **Git** con el comando:

```
git config
```

- Se recomienda establecer una identidad en **Git**, para ello se usan los comandos:

```
git config user.name
```

```
git config user.email
```

- Usando el flag “**--global**” podemos establecer la configuración de forma global y realizarla una sola vez.

```
git config --global user.name “devf”
```



Comandos básicos de Git

git init -> Inicializamos repositorio

git clone (url) -> Clonar repositorio existente

git status -> Nos muestra un desglose los archivos agregados y modificados

git add (nombre archivo) -> Agregamos archivos al staging area

git commit -m "Comentario" -> Agregamos comentario sobre el cambio hecho

git push -> Enviamos cambios a repositorio remoto

git checkout -b (nombre) -> Salimos de la rama actual creando una nueva

git checkout (nombre) -> Salimos de la rama actual a una existente

git merge (rama) -> Unimos cambios de una rama

git pull -> Obtenemos cambios más recientes de la rama

