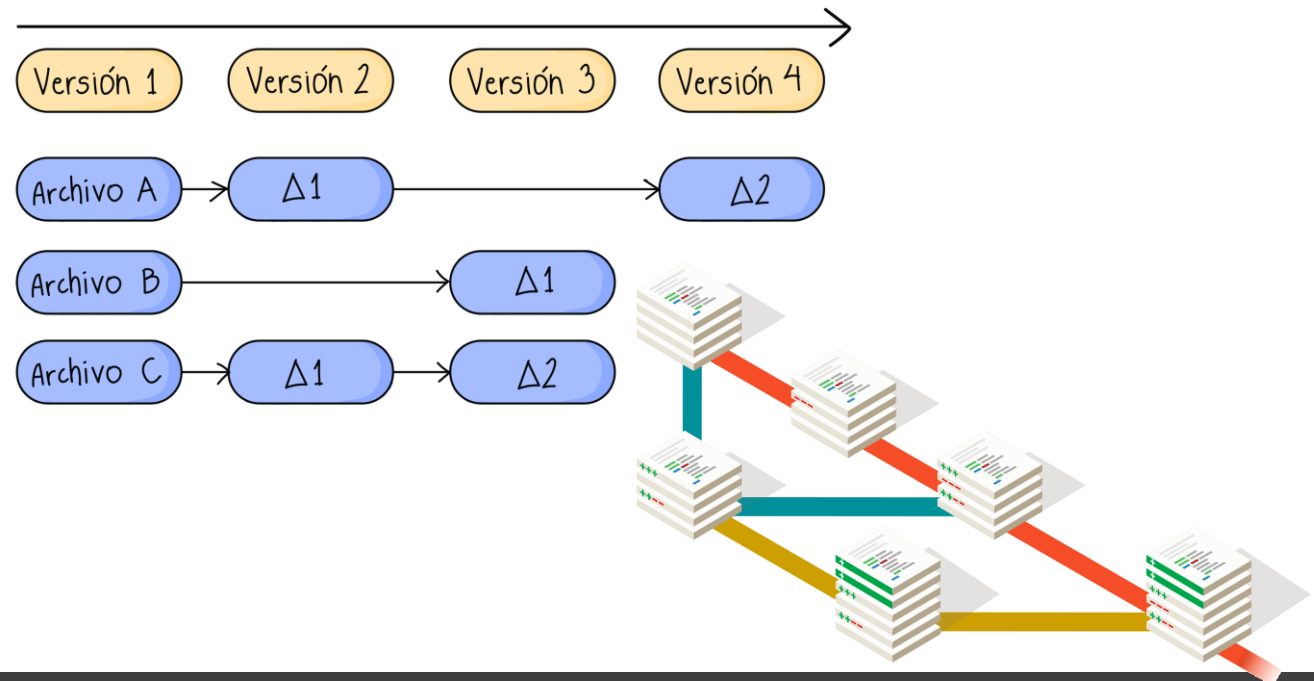
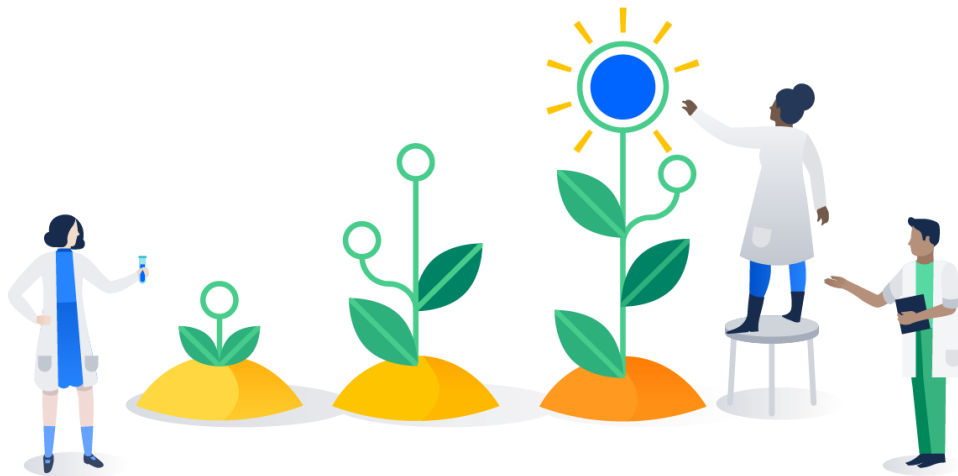




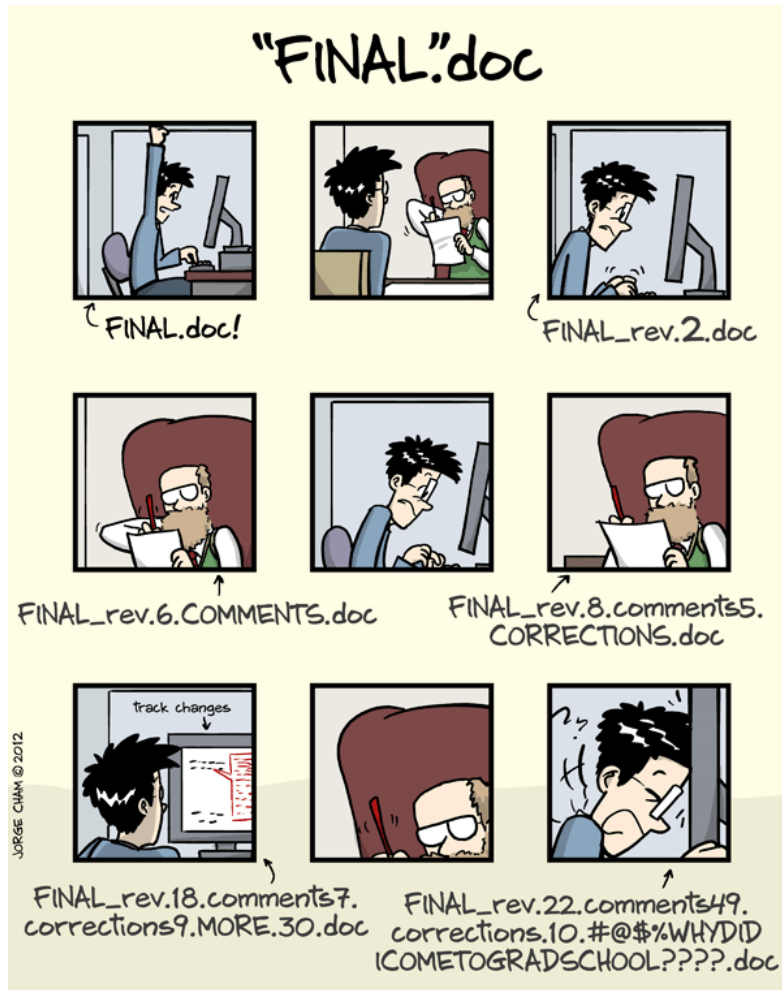
GitHub

# ¿Qué es Git?

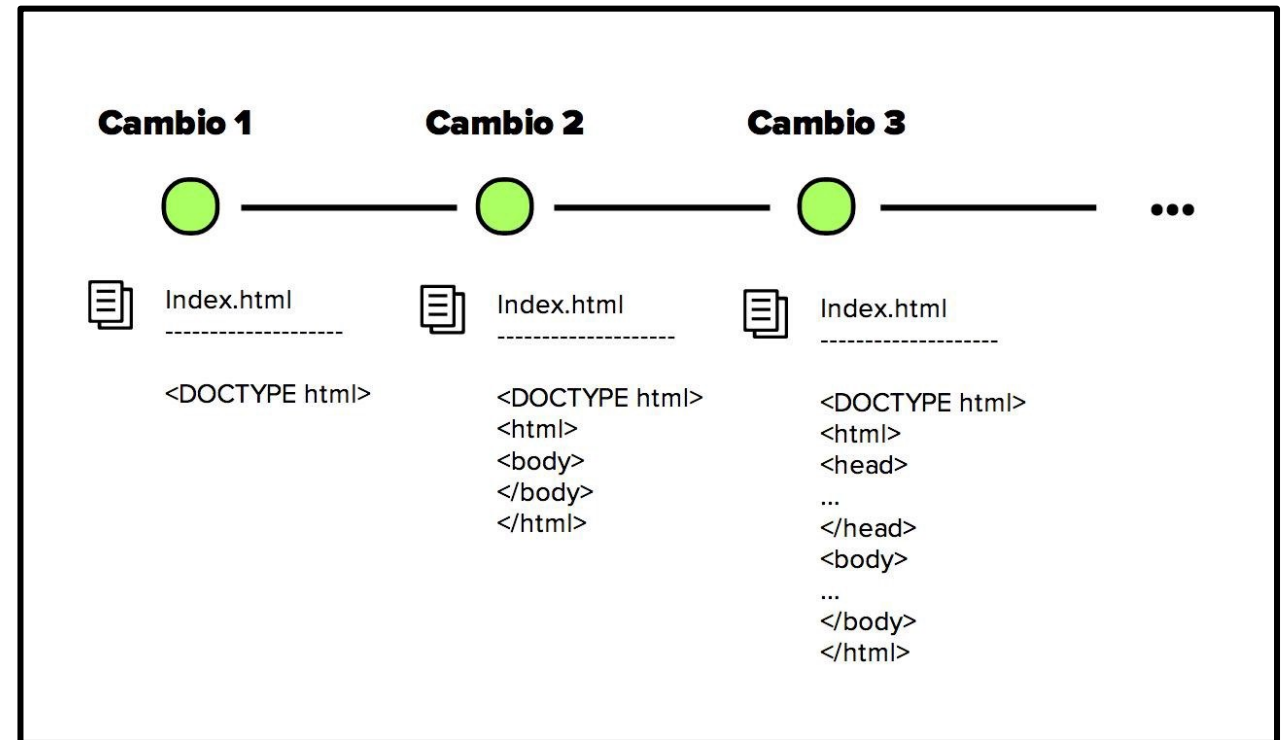
Es un sistema de control de versiones (VSC), su propósito es llevar registro de los cambios en archivos de computadora y coordinar el trabajo que varias personas realizan sobre archivos compartidos.



# Sistemas de Control de Versiones



VS



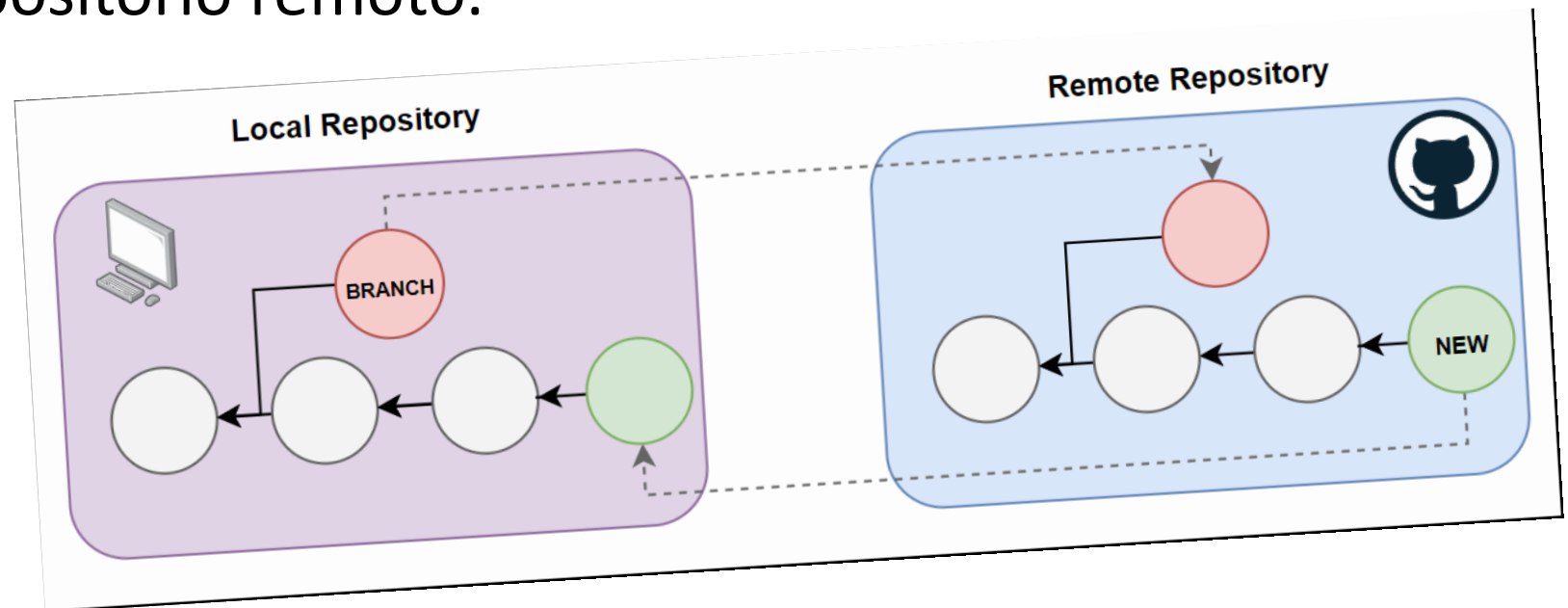
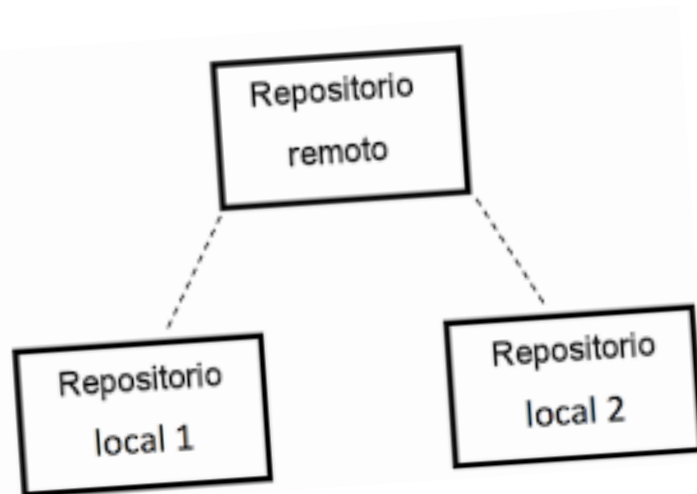
diseñado por Linus Torvalds, el creador de Linux.

# ¿Qué es Git?

Existe la posibilidad de trabajar de forma remota y una opción es GitHub.

## Características Principales de Git

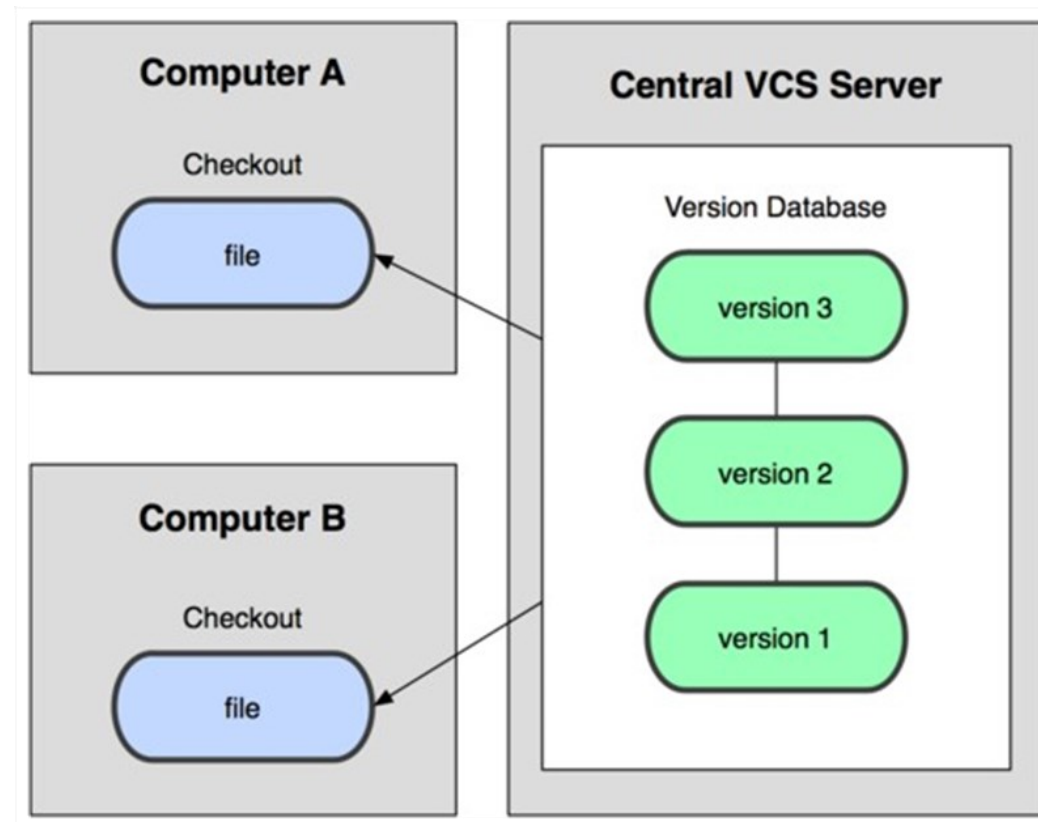
Repositorio local y repositorio remoto.



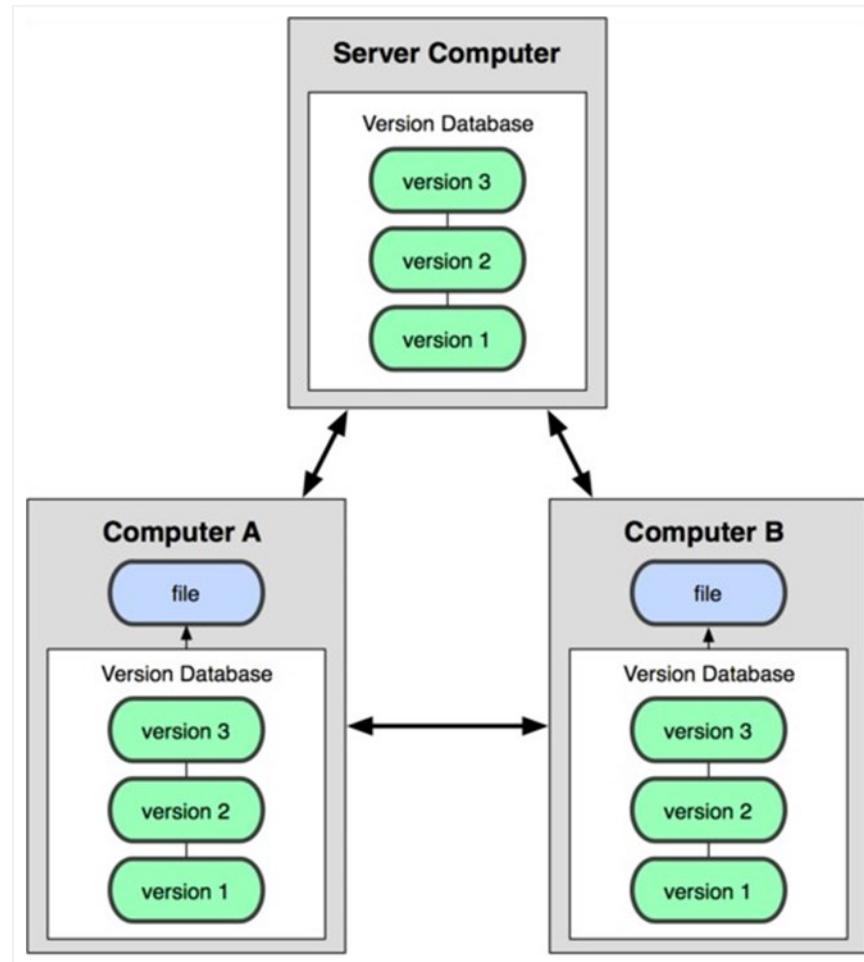
# ¿Por qué un Sistema de Control de Versiones?

- No trabajamos solos.
- Si modificamos un proyecto directamente, no témenos constancia de cómo era antes de los cambios.
- Seguridad.

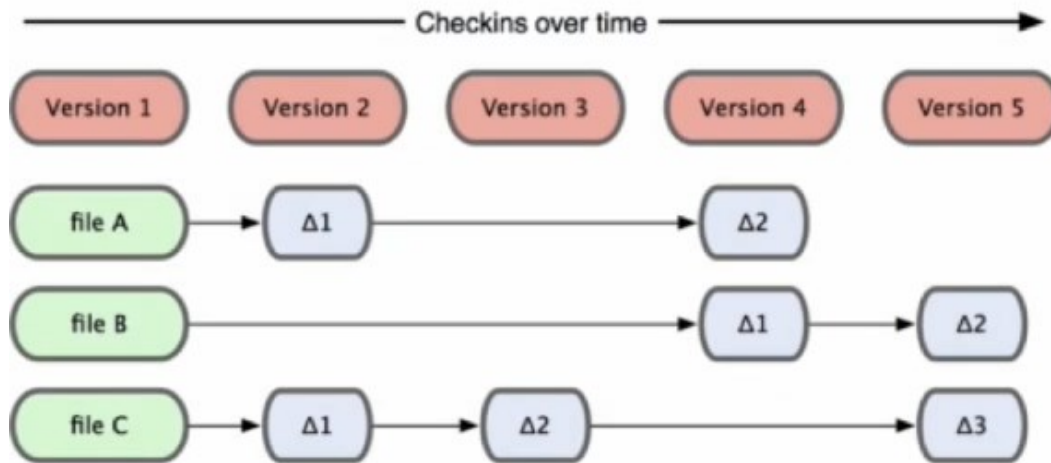
# Sistemas Centralizados



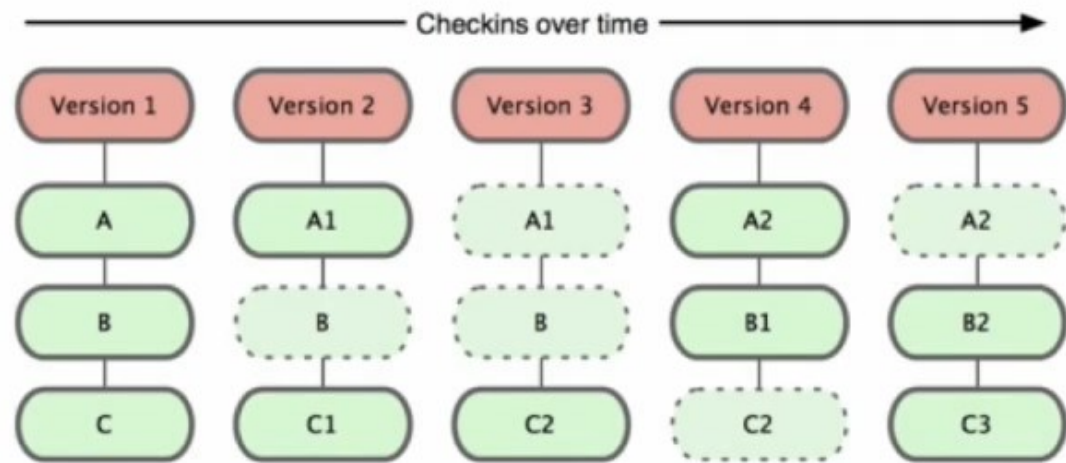
# Sistemas Distribuidos



# Centralizados VS Distribuidos



Diferencias



Instantáneas



# Rama o branch

- Son líneas paralelas de nuestro propio repositorio.
- Se crean ramas para que en la principal el código esté limpio.
- Si la rama funciona correctamente la añadimos a la principal (merged branch).



# Flujo de trabajo de GIT

## Working Directory

Carpeta local donde trabajamos y editamos los archivos.

## Staging Area

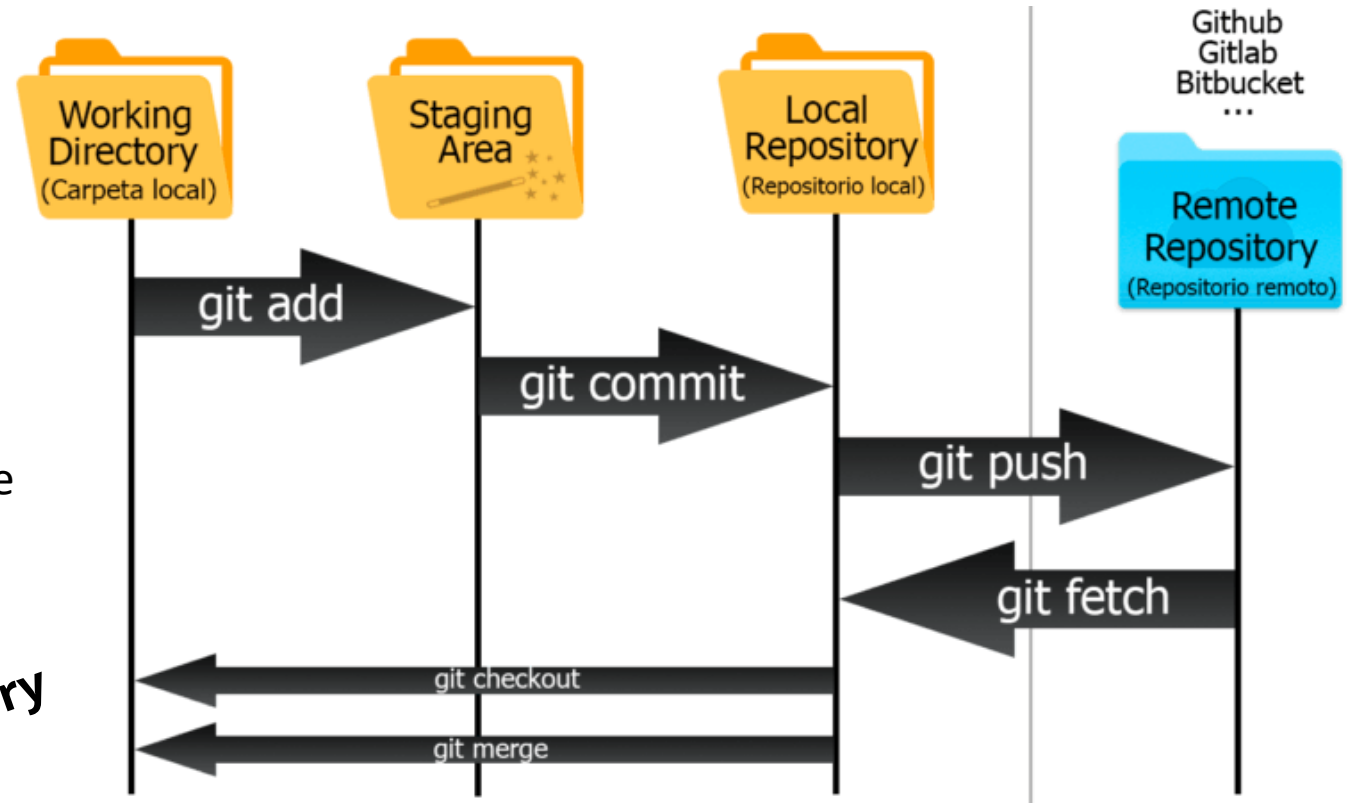
Lugar donde añadimos archivos cuando al terminar de editarlos.

## Remote repository

Repositorio remoto online.

## Local repository

Repositorio de nuestro ordenador.



# Flujo de trabajo de GIT

**git add**

Añade los archivos nuevos o modificados al área staging

**git commit**

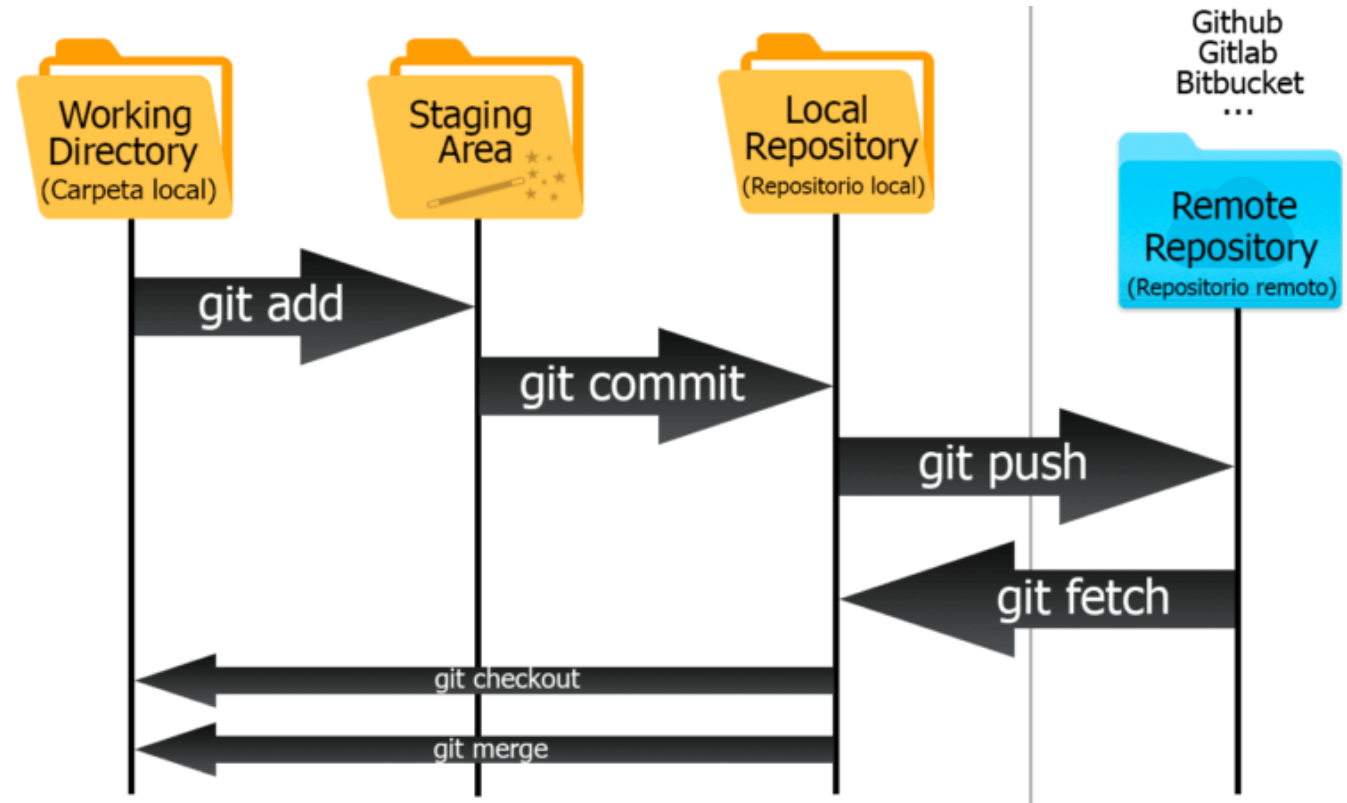
Registra o toma instantáneas del archivo de forma permanente en el historial de versiones.

**git push**

Envía los cambios confirmados de la rama correspondiente a su repositorio remoto.

**git fetch**

Descargar objetos y referencias del repositorio remoto



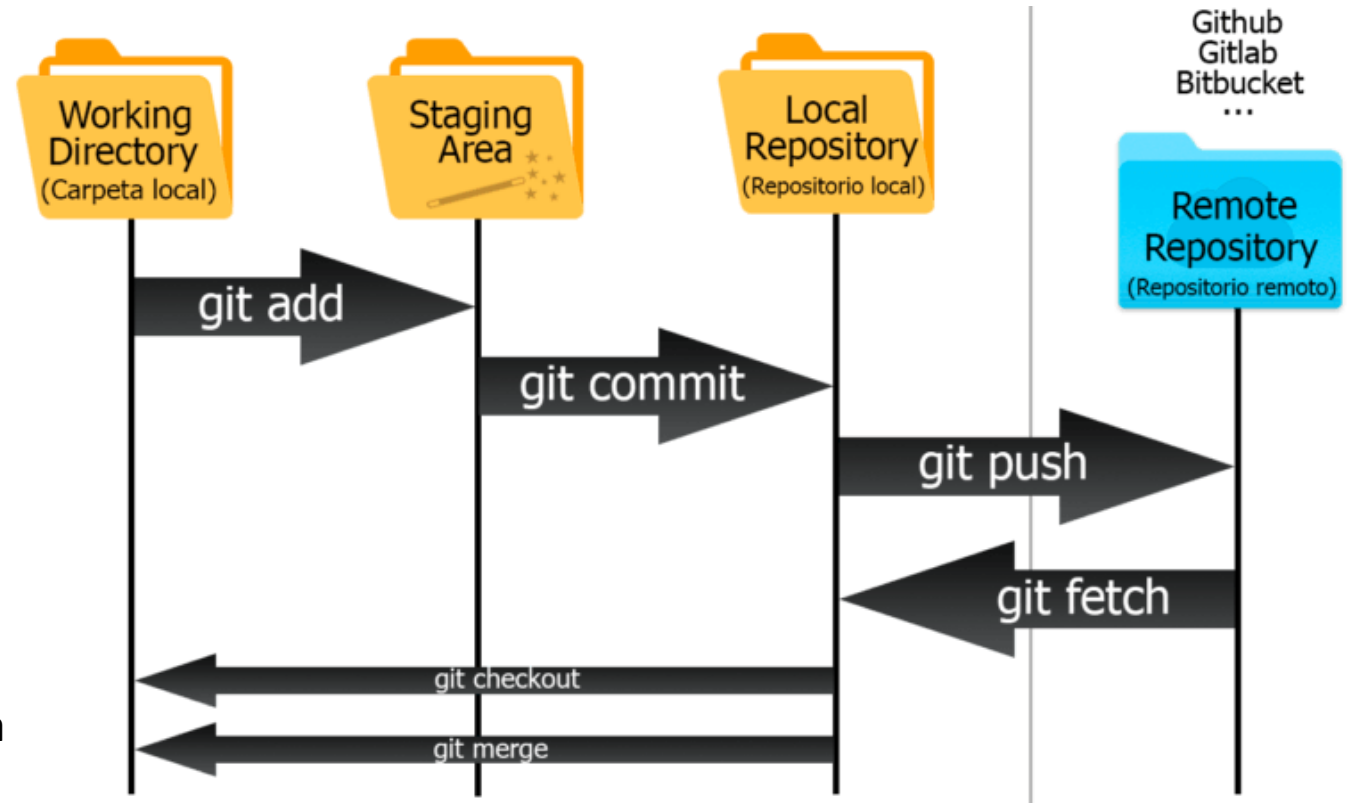
# Flujo de trabajo de GIT

**git checkout**

Cambiar de una rama a otra

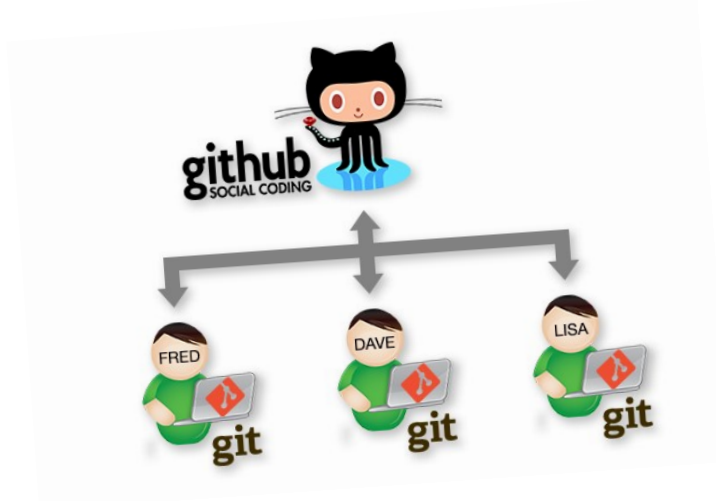
**git merge**

fusiona el historial de la rama especificada en la rama actual.



# ¿Qué es GitHub?

- Es una plataforma de desarrollo colaborativo para alojar proyectos (en la nube) utilizando el sistema de control de versiones Git.
- Repositorios externos (Github, gitlab, bitbucket...) para tener nuestro código disponible en cualquier parte.



# Términos al Crear un repositorio de Github

**README**

Archivo de documentación de repositorio.

**.gitignore**

Archivo que permite omitir archivos. Específicos.

**licencia**

especificar una licencia de código abierto.



# Razones para usar Github

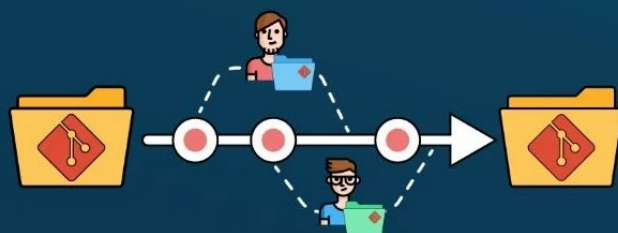
- Trabajo colaborativo.
- Volver a una versión anterior.
- Facilita el compartir archivos en un proyecto
- Creación de ramas y fusiones
- Permite generar flujos de trabajo
- Trazabilidad
- Resolución de conflictos
- Revertir y deshacer los cambios en el código fuente
- Copia de seguridad externa del código fuente



# ¿SABES QUÉ ES GIT?



Es un sistema de **control de versiones**. Lleva un registro de todos los **cambios y avances** de tu proyecto.



## ● GIT TRABAJA CON RAMAS

Ayuda a que **varias personas trabajen** en un mismo proyecto y pueden realizar **modificaciones sin afectar a los demás** archivos. Una vez que estén listos los cambios se **fusionan con la rama principal**.

Prof. Beto Quiroga



Todo desarrollador sin importar el lenguaje **debe dominar Git**.

v1 — v2 — v3

● Funciona como una **máquina de tiempo**, puedes ir al pasado de tu código o volver al presente.

● Github es un servicio que te ayuda a **almacenar tu proyecto en la nube**, además existen otros servicios como **Gitlab** o **Bitbucket**.