

1. Git

Git es un sistema de control de versiones distribuido que permite a los desarrolladores hacer un seguimiento del flujo de trabajo en un proyecto, ya sea individual o colaborativo. Es de código abierto (con mantenimiento activo), multiplataforma y compatible con variedad de IDEs esto, sumado a sus características, lo posicionan como la herramienta más empleada para el control de versiones.

Algunas de las características que hacen a su popularidad son:

- **Es Distribuido:** En lugar de tener un único espacio para todo el historial de versiones del software, en Git la copia de trabajo del código de cada desarrollador es un repositorio local que puede albergar el historial completo de todos los cambios.
- **Trabaja con Ramas (branches):** Emplea un sistema de trabajo con ramas que lo hace especialmente potente. Dichas ramas están destinadas para hacer cambios y/o mejoras divergentes del proyecto principal (cada una puede tener una línea de progreso diferente de la principal). Esto permite agregar esos cambios al código cuando sea oportuno.
- **Rendimiento, seguridad y flexibilidad:** Linus Torvalds (creador del kernel de Linux) cimentó Git sobre estas bases.

2. Comandos de Git

A continuación se destacan algunos de los comandos más importantes de Git.

Comandos básicos (de consola)	Para la configuración	Para el control de versiones
<code>pwd</code> (print working directory) Muestra la dirección actual.	<code>git --version</code> Indica la versión de Git.	<code>git init</code> Inicializa un nuevo repositorio
<code>ls</code> (list) Lista los archivos y carpetas.	<code>git config --global user.name</code> Define el nombre de usuario.	<code>git status</code> Estado actual del repositorio
<code>ls -a</code> Incluye arch. y carp. ocultas.	<code>git config --global user.email</code> Define el email del usuario.	<code>git add</code> Hace un seguimiento.
<code>cd</code> Entra a una carpeta.	<code>git config --global core.editor</code> Define el editor de código.	<code>git commit -m</code> Compromete los cambios.
<code>cd ..</code> Vuelve a la carpeta anterior.	<code>git config --global -e</code> Muestra la configuración.	<code>git checkout -b</code> Crea una nueva rama.
<code>mkdir</code> Crea una nueva carpeta.	<code>git config -h</code> Opciones de configuración.	<code>git branch</code> Muestra la rama actual.
<code>touch</code> Crea un nuevo archivo.	<code>git config --list</code> Comprobar la configuración.	<code>git branch -M</code> Cambia el nombre de la rama actual.

		<code>git push -u origin main</code> Sube los cambios al repositorio remoto.
		<code>git pull</code> Extrae contenido de un rep.
		<code>git clone</code> Clona un repositorio.

3. GitHub

GitHub es una plataforma de gestión y organización de proyectos basada en la nube que incorpora las funciones de control de versiones de Git. Ésta plataforma permite a los desarrolladores colaborar y realizar cambios en proyectos compartidos, a la vez que mantienen un seguimiento detallado de su progreso. Es decir que todos los usuarios de GitHub pueden rastrear y gestionar los cambios que se realizan en el código fuente en tiempo real, a la vez que tienen acceso a todas las de más funciones de Git disponibles.