


Crear el proyecto de git

Ahora que haz comprendido la importancia de Git, es momento de instalar esta poderosa herramienta en tu computadora para poder comenzar a utilizarlo en tu proyecto.

Actividad: Instalar Git

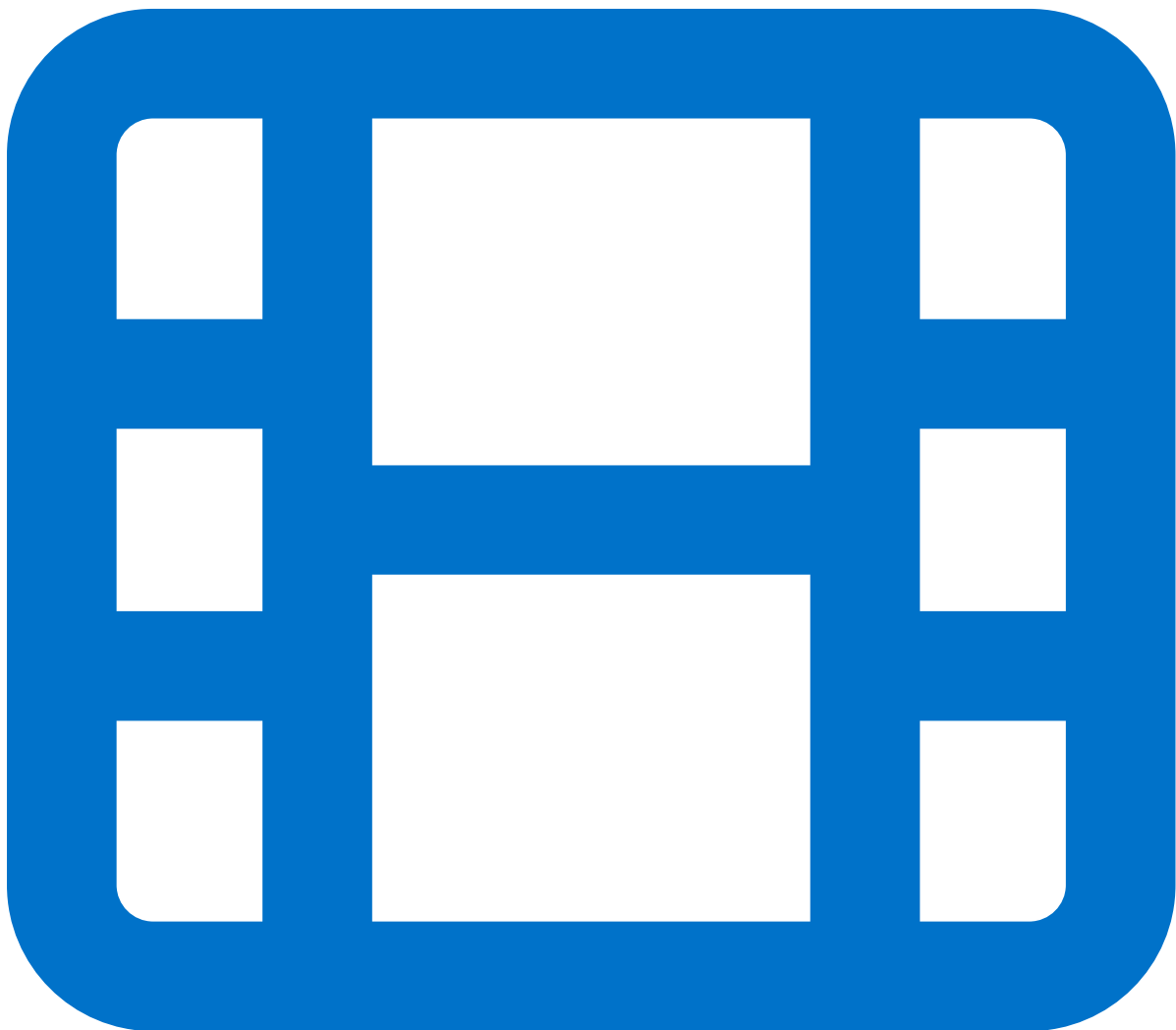
Pasos para instalar Git:

- Ingresa al sitio web oficial de Git 
<https://git-scm.com/downloads>
- Selecciona la versión de Git correspondiente a tu sistema operativo (Windows, MacOS o Linux).
- Haz clic en el enlace de descarga para iniciar la descarga del instalador.
- Una vez que descargado, ejecuta el archivo de instalación.
- Haz clic en “Next” y continúa haciendo clic en “Next” hasta llegar al botón de “Install”.

- Haz clic en “Install” y, por último, en “Finish”.

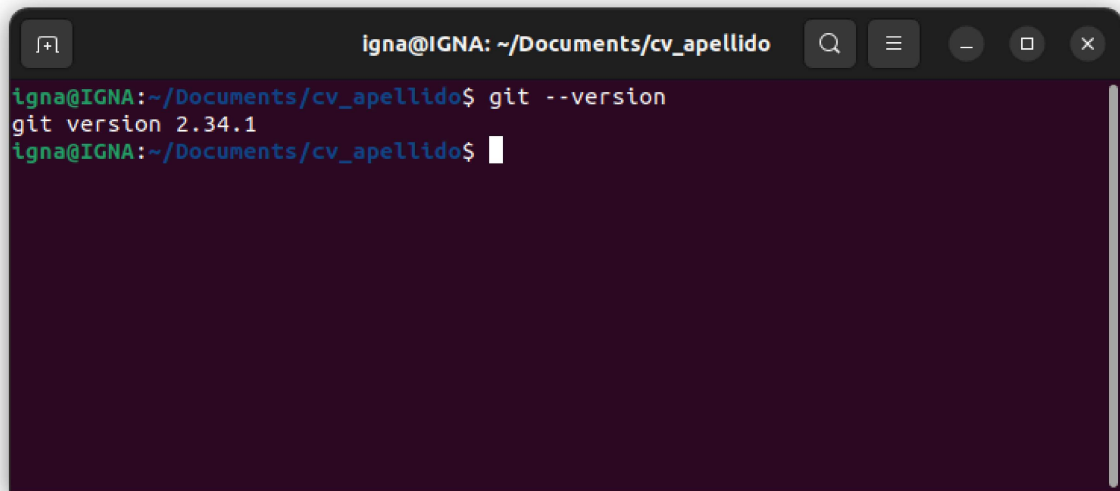
A continuación, te compartimos el video de instalación, o bien, puedes optar por seguir el paso a paso descrito abajo del mismo:

👉 Si necesitas ayuda para realizar la instalación, te dejamos un video para guiarte



Instalación de Git | Introducción a Git | Sistema de control de versiones

¡Listo! 🙌 Una vez que la instalación haya finalizado, recomendamos reiniciar la computadora y verificar si Git se ha instalado correctamente abriendo la terminal y ejecutando el comando: `git --version`.



```
igna@IGNA: ~/Documents/cv_apellido
igna@IGNA:~/Documents/cv_apellido$ git --version
git version 2.34.1
igna@IGNA:~/Documents/cv_apellido$
```



Actividad: Configurar Git

Después de instalar Git, es importante realizar una configuración inicial utilizando el comando “`git config`”. Esta configuración es necesaria ya que Git la requerirá para poder identificar al autor de los cambios y asegurar la trazabilidad de los cambios.

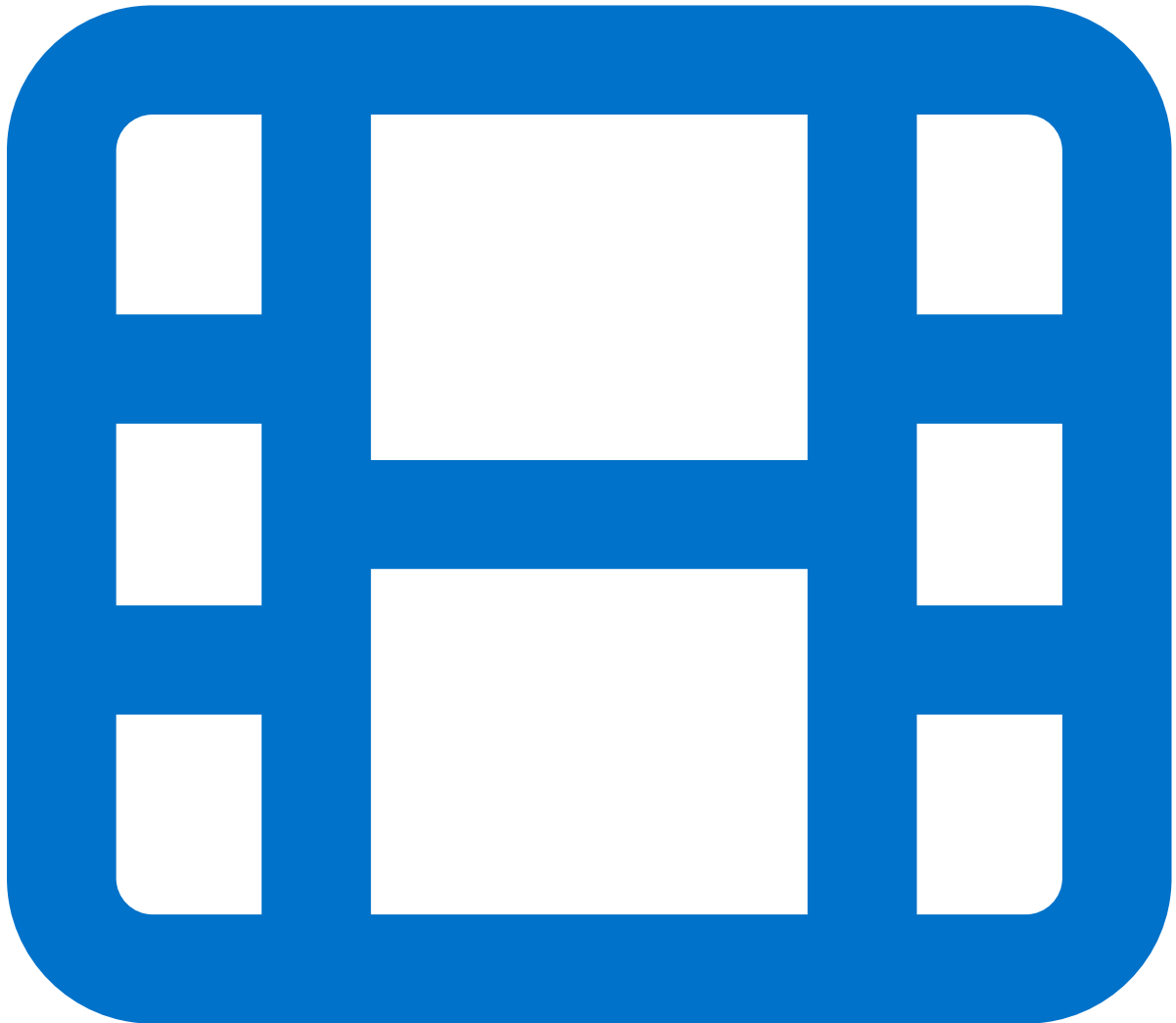
La configuración inicial consiste en establecer:

- Tu nombre de usuario, con el comando: **`git config --global user.name “Tu nombre”`**
- Tu dirección de correo, con el comando: **`git config --global user.email “Tu email”`**

Desde la consola, como por ejemplo:

```
git config --global user.name "nickname user"  
git config --global user.email "nickname@mail.com"
```

👉 Si necesitas ayuda, te dejamos un video para guiarte



[Introducción a Git | Configurar Git | Egg](#)

Ahora que has realizado la configuración inicial, estás listo para comenzar a gestionar tus proyectos en Git.
¡Adelante!



Actividad: Crear un repositorio de Git “Local”

Teniendo ya instalado y configurado Git en nuestro ordenador, pasaremos a entender cómo crear un repositorio local.

Un repositorio es un espacio centralizado donde se almacena, organiza y mantiene la información.

Es “la carpeta” o espacio donde guardaremos el proyecto para más adelante compartirlo con otros colaboradores a través de un repositorio en la nube (como por ejemplo, en Github).

1. Crea una nueva carpeta en tu ordenador con el nombre de “**profile**”, hacer clic derecho sobre la misma y seleccionar la opción adecuada para abrir esa carpeta desde una consola de comandos o terminal (en windows suele ser la opción “Git Bash Here” y en linux “Open in terminal”).

2. Escribe en la terminal el siguiente comando y luego dar un “enter”:

```
git init
```

Al ejecutar “git init”, estás creando un nuevo repositorio Git vacío (o si ya existe lo estarías reiniciando). Si se creó correctamente deberías ver el mensaje: “**Initialized empty Git repository in /ruta/a/tu/directorio/.git/**”.

```
igna@IGNA: ~/Documents/cv_apellido
igna@IGNA:~/Documents/cv_apellido$ git init
hint: Using 'master' as the name for the initial branch. This default branch nam
e
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/igna/Documents/cv_apellido/.git/
igna@IGNA:~/Documents/cv_apellido$
```

3. Escribe el siguiente comando y luego dar un “enter”:
`git status`

Con “`git status`” estamos chequeando nuestro repositorio.
La consola informará el estado actual del mismo.

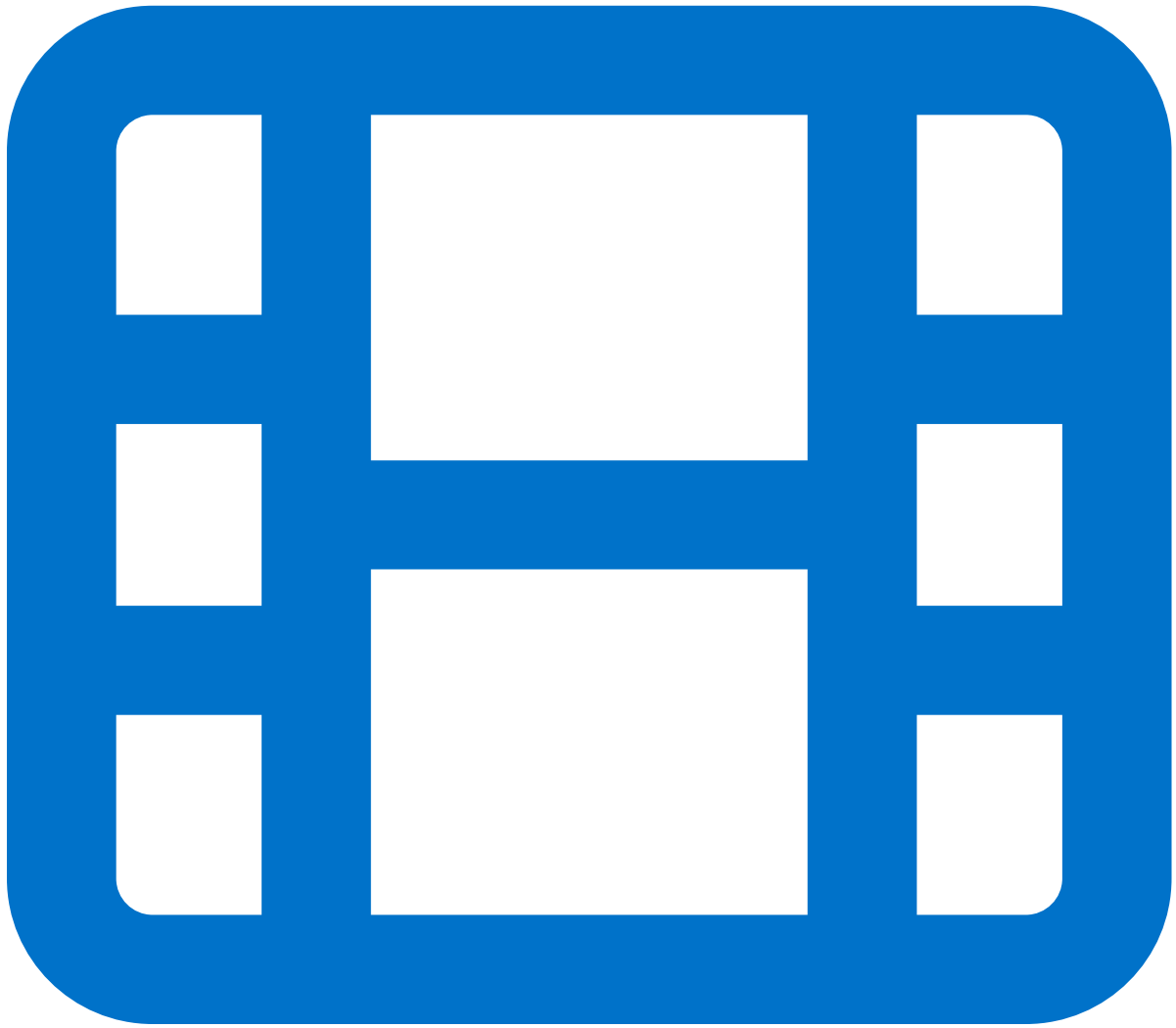
```
igna@IGNA: ~/Documents/cv_apellido
igna@IGNA:~/Documents/cv_apellido$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
igna@IGNA:~/Documents/cv_apellido$
```

En esta ocasión observamos que estamos en nuestro directorio de trabajo (“**working directory**”), posicionados en la rama (“**branch**”) principal del mismo (“**master**” o también puede figurar como “**main**”).

👉 Si necesitas ayuda, te dejamos un video para guiarte



[Introducción a Git](#) | [Git init](#) | [Egg](#)

Actividad: Crear primera versión

Ahora estamos en condiciones de empezar un proyecto.

1. Con la consola de comandos abierta en la carpeta “**profile**”, ejecutar:
`touch README.md`

Este comando crea un nuevo archivo llamado “**README.md**” en carpeta “**profile**”.

Este archivo está en tu directorio de trabajo. Esto significa que Git sabe que el archivo está ahí, pero no está siguiendo los cambios del mismo todavía.

2. Ejecutar el comando:

```
git add README.md
```

Desde este momento, el archivo está en el área de preparación (“staging area”) y desde este punto el archivo y cada cambio que tenga es rastreado por git.

Para verificar si el archivo cambió de ubicación” (del working al staging) ejecutar “git status”. El archivo debería figurar debajo de “Changes to be committed”.

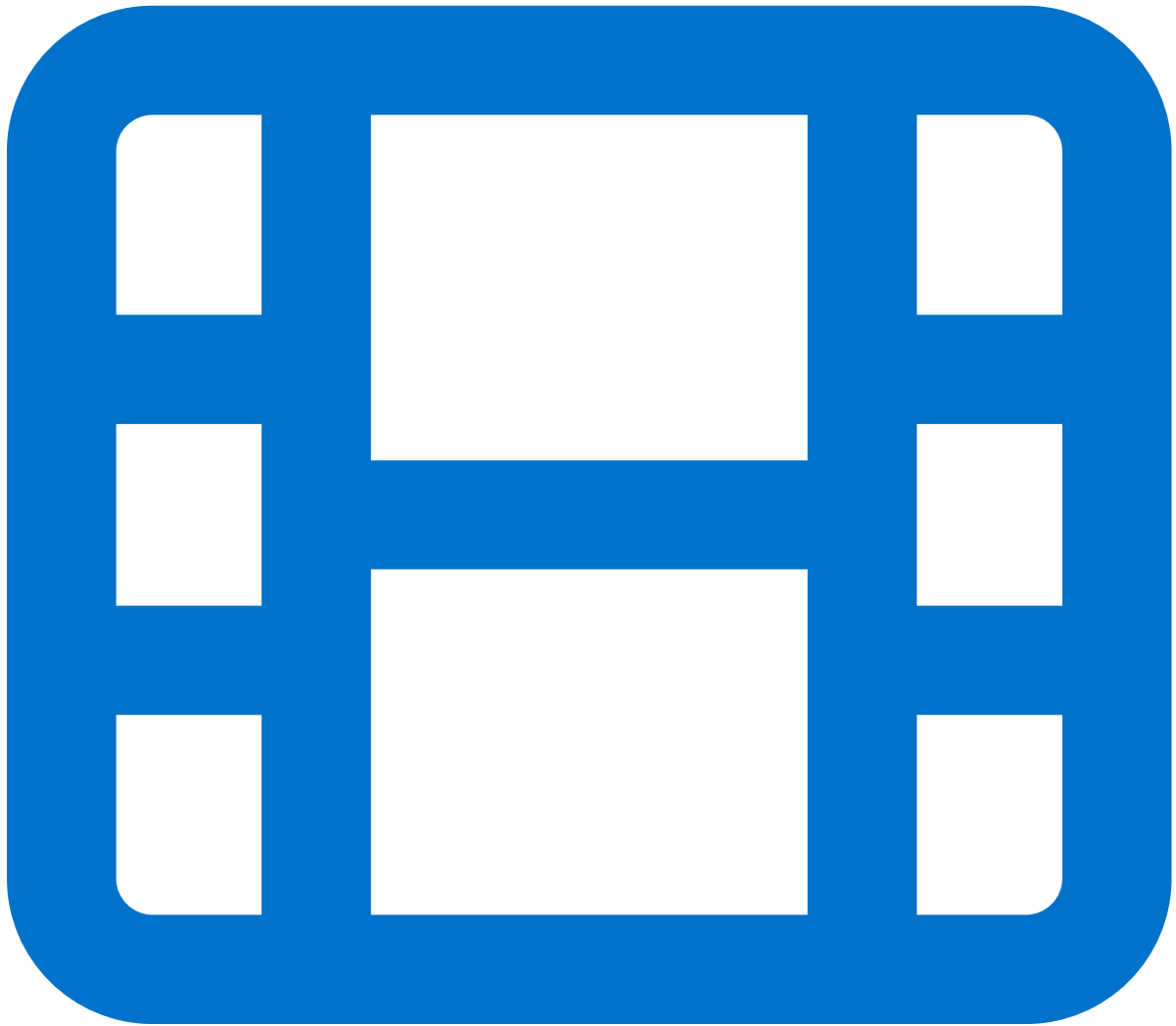
3. Finalmente, ejecutar el siguiente comando:

```
git commit -m "creacion readme"
```

Con este comando, realizamos el primer envío, la primera publicación o versión del proyecto.

Este comando toma todos los cambios que están en el área de preparación y los guarda en el repositorio local. Git sabe que este es un punto de cambio que quieres registrar

👉 Si necesitas ayuda, te dejamos un video para guiarte



Introducción a Git | Primer commit | Egg

¡IMPORTANTE! Ten en cuenta

- *El mensaje del commit debe ser lo más breve y descriptivo posible.*
- *Para verificar si el archivo cambió de ubicación” (del staging al local repository) ejecutar “git status”. La consola debería informar “Nothing to commit, working tree clean”.*
- *Antes de realizar un nuevo commit, es necesario crear, modificar o eliminar*

archivos de tu proyecto.

- *No nos olvidemos los pasos: primero hay que mover el o los archivos desde el directorio al área de preparación (git add) y luego desde este área al repositorio local (git commit)*