

Primero necesitamos descargar la herramienta de git desde la terminal de linux para poder subir un archivo o proyecto con el comando

`sudo apt-get install git`



A terminal window titled "serbero@serbero-HP-Pavilion-dm4-Notebook-PC: ~" with a menu bar containing "Archivo", "Editar", "Ver", "Buscar", "Terminal", and "Ayuda". The command prompt shows the user "serbero@serbero-HP-Pavilion-dm4-Notebook-PC" at the tilde (~) directory, followed by the command "sudo apt install git" with a cursor at the end.

Te pedira la contraseña de tu usuario root para poder descargarlo la ingresas y comenzara a descargarlo



A terminal window titled "serbero@serbero-HP-Pavilion-dm4-Notebook-PC: ~" with a menu bar containing "Archivo", "Editar", "Ver", "Buscar", "Terminal", and "Ayuda". The command prompt shows the user "serbero@serbero-HP-Pavilion-dm4-Notebook-PC" at the tilde (~) directory, followed by the command "sudo apt install git". Below this, the prompt "[sudo] contraseña para serbero: \*\*\*\*\*" is shown with a cursor at the end of the asterisks.

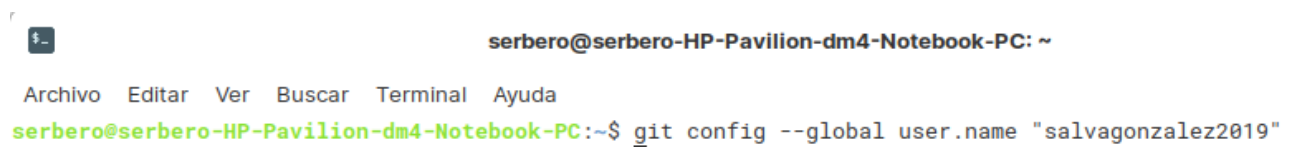
te mostrara un mensaje sobre el tamaño del archivo y le daras un S para continuar o un enter

```
serbero@serbero-HP-Pavilion-dm4-Notebook-PC:~$ sudo apt install git
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
El paquete indicado a continuación se instaló de forma automática y ya no es necesario.
  libllvm7
Utilice «sudo apt autoremove» para eliminarlo.
Se instalarán los siguientes paquetes adicionales:
  git-man liberror-perl
Paquetes sugeridos:
  git-daemon-run | git-daemon-sysvinit git-doc git-el git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
Se instalarán los siguientes paquetes NUEVOS:
  git git-man liberror-perl
0 actualizados, 3 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 4 733 kB de archivos.
Se utilizarán 33.9 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n]
```

## CONFIGURACION INICIAL DE GIT

El primer comando ingresaras el nombre de tu usuario antes ya echo en la pagina oficial

```
git config --global user.name "nombre_usuario"
```



The screenshot shows a terminal window with a title bar containing a file icon, the text "serbero@serbero-HP-Pavilion-dm4-Notebook-PC: ~", and navigation buttons: "Archivo", "Editar", "Ver", "Buscar", "Terminal", and "Ayuda". The terminal content shows the command `git config --global user.name "salvagonzalez2019"` being entered at the prompt `serbero@serbero-HP-Pavilion-dm4-Notebook-PC:~$`.

```
git config --global user.email "email_id"
```

```
serbero@serbero-HP-Pavilion-dm4-Notebook-PC:~$ git config --global user.email "salvarm077@hotmail.com"
```

## Creación de un repositorio local

Lo primero es crear una carpeta en tu computadora, que servirá como un repositorio local. Para ello, basta ejecutar el siguiente comando:

```
git init nombrerepositorio
```

```
serbero@serbero-HP-Pavilion-dm4-Notebook-PC:~$ git init prueba
Inicializado repositorio Git vacío en /home/serbero/prueba/.git/
serbero@serbero-HP-Pavilion-dm4-Notebook-PC:~$ █
```

Este comando crea la carpeta nombrerepositorio. A su vez, la sub-carpeta .init hace que nombrerepositorio sea reconocido como un repositorio local de Git.

Si se crea el repositorio con éxito, aparecerá una línea similar a la siguiente:

```
Initialized empty Git repository in /home/tu_usuario/nombrerepositorio/.git/
```

Acto seguido, hay que entrar a la carpeta :

```
cd nombrerepositorio
```

```
serbero@serbero-HP-Pavilion-dm4-Notebook-PC:~$ cd arquitectura1.1
```

ahora el comando “git add” se puede utilizar para agregar cualquier número de archivos y carpetas al índice. Para agregar todos los cambios, sin especificar el nombre de los archivos, -A AGREGA TODOS LOS ARCHIVOS

```
serbero@serbero-HP-Pavilion-dm4-Notebook-PC:~/arquitectura1.1$ git add -A
serbero@serbero-HP-Pavilion-dm4-Notebook-PC:~/arquitectura1.1$ git status
En la rama master
```

No hay commits todavía

Cambios a ser confirmados:

(usa "git rm --cached <archivo>..." para sacar del área de stage)

```
nuevo archivo: 3051_JIMENEZ_GONZALEZ_SALVADOR_discoshdd&SDD.pdf
nuevo archivo: README
```

## Guardar los cambios realizados en el índice

Una vez añadidos todos los archivos, es posible dejar un registro de estos cambios haciendo lo que en la jerga se llama un “commit”. Esto significa que ya se ha terminado de agregar o modificar archivos y que los cambios pueden ser subidos al repositorio remoto de Github. Para ello, hay que ejecutar el siguiente comando:

```
git commit -m "mensaje"
```

“mensaje” puede ser cualquier mensaje que describa brevemente los cambios en cuestión, por ejemplo: “agregué tal funcionalidad” o “corregí tal cosa”, etc.

```
serbero@serbero-HP-Pavilion-dm4-Notebook-PC:~/arquitectura1.1$ git commit -m "tarea 1"
[master (commit-raiz) a0264a8] tarea 1
 2 files changed, 2 insertions(+)
 create mode 100644 3051_JIMENEZ_GONZALEZ_SALVADOR_discoshdd&SDD.pdf
 create mode 100644 README
```

## Creación de un repositorio en GitHub

El nombre del repositorio debe ser el mismo que el repositorio del sistema local. En este caso, será “MyTest”. Para ello, antes que nada, hay que iniciar sesión en Github. Luego, hay que hacer clic en el signo más (+) en la esquina superior derecha de la página y seleccionar la opción “crear nuevo repositorio”. Finalmente, hay que rellenar los datos y hacer clic en el botón “crear repositorio”.

Una vez hecho esto se creará el repositorio y será posible subir el contenido del repositorio local en el repositorio GitHub. Para conectarse al repositorio remoto en GitHub hay que ejecutar el comando:

```
git remote add origin https://github.com/user_name/nombrerepositorio.git
```

```
create mode 100644 README
```

```
serbero@serbero-HP-Pavilion-dm4-Notebook-PC:~/arquitectura1.1$ git remote add origin https://github.com/salvagonzalez2019/arquitectura.git
```

## Empujar archivos del repositorio local al repositorio GitHub

El paso final es empujar el contenido del repositorio local hacia el repositorio remoto, mediante el comando:

```
git push origin master
```

Sólo resta introducir las credenciales de inicio de sesión (nombre de usuario y contraseña).

Esto subirá todo el contenido de la carpeta MyTest (repositorio local) a GitHub (repositorio externo). Para los proyectos subsiguientes ya no hará falta seguir estos pasos desde cero. En cambio, podrás empezar desde el paso 3 directamente. Por último, no te olvides que los cambios estarán disponibles desde la página web de Github

```
serbero@serbero-HP-Pavilion-dm4-Notebook-PC:~/arquitectura1.1$ git push -u origin master
Username for 'https://github.com': salvagonzalez2019
Password for 'https://salvagonzalez2019@github.com':
Contando objetos: 4, listo.
Delta compression using up to 4 threads.
Comprimiendo objetos: 100% (4/4), listo.
Escribiendo objetos: 100% (4/4), 30.36 KiB | 30.36 MiB/s, listo.
Total 4 (delta 0), reused 0 (delta 0)
To https://github.com/salvagonzalez2019/arquitectura.git
 * [new branch]      master -> master
Rama 'master' configurada para hacer seguimiento a la rama remota 'master' de 'origin'.
```