

## Actividad Evaluable

Módulo	Servicios de red
Nombre y Apellidos:	Diego Cota
Nombre y Apellidos:	Eva Pulla
Nombre y Apellidos:	
Estas soluciones deben ser entregadas en PDF.	

**FECHA DE ENTREGA: --/--/21**

### Objetivos

---

El objetivo de esta actividad individual es la de repasar, asentar y adquirir un mayor conocimiento de lo impartido en clase.

El archivo debe nombrarse como: **nombre\_apellido1\_apellido2.PDF**

Ej: Belen\_Vargas\_Sabater.pdf

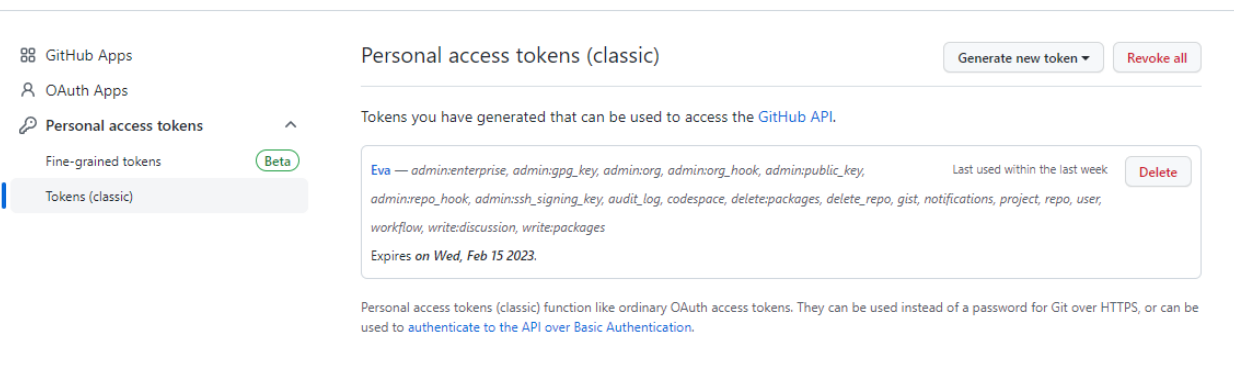
## Cuestiones

**EJERCICIO 1: Práctica:** Realizar una guía práctica a modo de tutorial, sobre el uso de GitHub, algunos de los elementos que debe cubrir son:

- Creación de repositorio y conexión con repositorio personal de GitHub mediante terminal con ejemplos
- Push Y pulls a los repositorios desde el terminal con ejemplos.
- Cada miembro del equipo debe realizar cambios en el repositorio propiedad de uno de los miembros, documenta el proceso.

**1º** Entramos en GitHub con nuestra cuenta, nos vamos a settings y abajo del todo encontraremos la opción Developer settings y entramos en esa opción. Una vez dentro de Developer settings seleccionamos Tokens (classic). Lo que hacemos en tokens es eliminar los que haya generados dando en Delete que aparece en color rojo.

Settings / Developer settings



Settings / Developer settings

Personal access tokens (classic) Generate new token ▼ Revoke all

Tokens you have generated that can be used to access the [GitHub API](#).

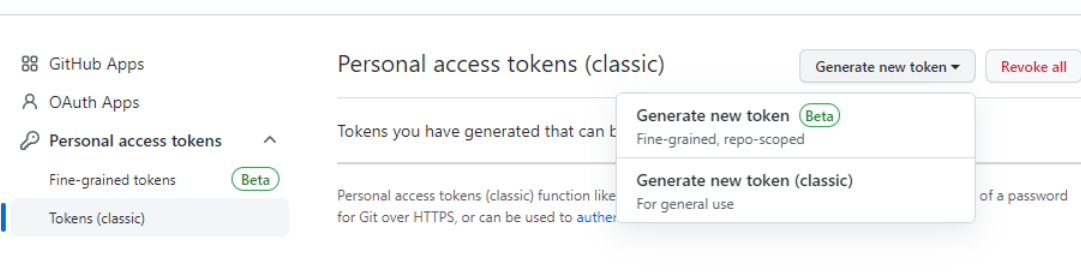
Token Name	Permissions	Last used	Action
Eva	admin:enterprise, admin:gpg_key, admin:org, admin:org_hook, admin:public_key, admin:repo_hook, admin:ssh_signing_key, audit_log, codespace, delete:packages, delete_repo, gist, notifications, project, repo, user, workflow, write:discussion, write:packages	Last used within the last week	Delete

Expires on Wed, Feb 15 2023.

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

**2º** Ahora una vez eliminados los tokens, generamos un nuevo token pinchando sobre Generate new token y escogemos la opción Generate new token (classic) ya que es mas sencilla y requiere menos datos.

Settings / Developer settings



Settings / Developer settings

Personal access tokens (classic) Generate new token ▼ Revoke all

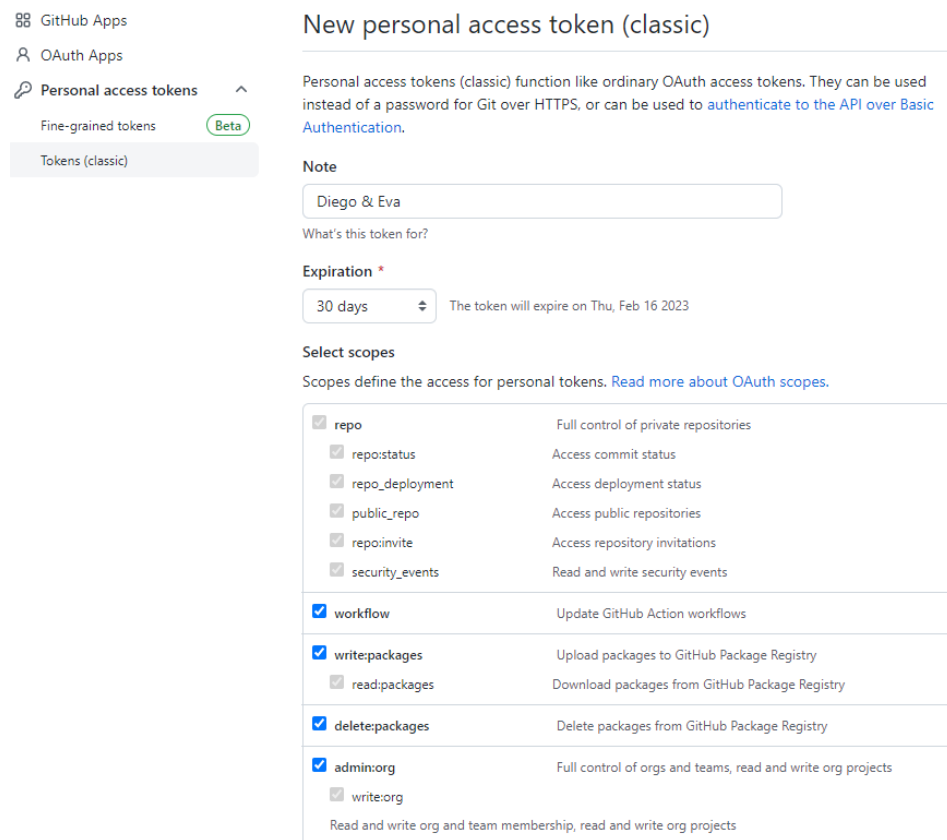
Tokens you have generated that can be used to access the [GitHub API](#).

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Generate new token (Beta) Beta  
Fine-grained, repo-scoped

Generate new token (classic)  
For general use

**3º** Añadimos una nota para identificar el token y marcamos todas las casillas para dar acceso a todo lo relacionado con el repositorio.



GitHub Apps

OAuth Apps

Personal access tokens <sup>Beta</sup>

Fine-grained tokens

Tokens (classic)

### New personal access token (classic)

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

**Note**

Diego & Eva

What's this token for?

**Expiration \***

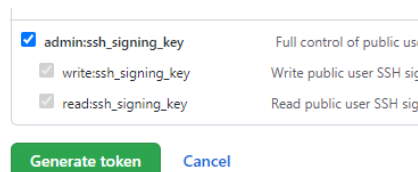
30 days The token will expire on Thu, Feb 16 2023

**Select scopes**

Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

<input checked="" type="checkbox"/> repo	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo_deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories
<input checked="" type="checkbox"/> repo:invite	Access repository invitations
<input checked="" type="checkbox"/> security_events	Read and write security events
<input checked="" type="checkbox"/> workflow	Update GitHub Action workflows
<input checked="" type="checkbox"/> write:packages	Upload packages to GitHub Package Registry
<input type="checkbox"/> read:packages	Download packages from GitHub Package Registry
<input checked="" type="checkbox"/> delete:packages	Delete packages from GitHub Package Registry
<input checked="" type="checkbox"/> admin:org	Full control of orgs and teams, read and write org projects
<input type="checkbox"/> write:org	Read and write org and team membership, read and write org projects

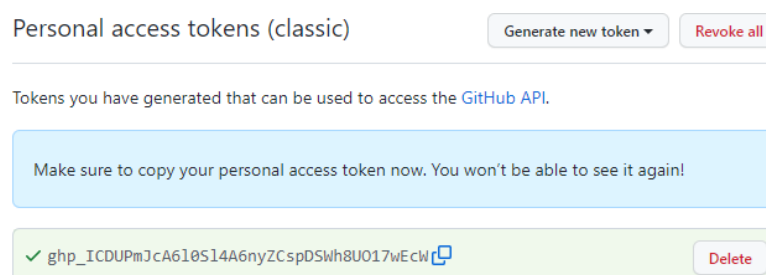
**4º** finalmente cuando ya hemos marcado todas las casillas damos a Generate token para que se nos genere.



<input checked="" type="checkbox"/> admin:ssh_signing_key	Full control of public user SSH keys
<input type="checkbox"/> write:ssh_signing_key	Write public user SSH sig
<input type="checkbox"/> read:ssh_signing_key	Read public user SSH sig

Generate token Cancel

**5º** Se muestra que ya hemos generado el token y guardamos muy bien el token (se recomienda copiarlo y pegarlo en un archivo de texto) ya que no volverá a mostrarlo.



Personal access tokens (classic) Generate new token Revoke all

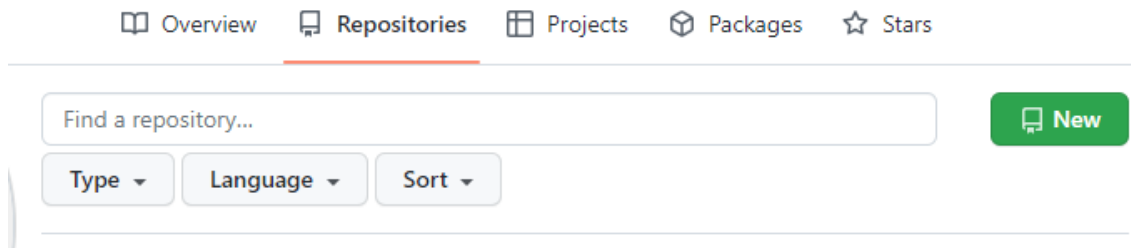
Tokens you have generated that can be used to access the [GitHub API](#).

Make sure to copy your personal access token now. You won't be able to see it again!

✓ ghp\_ICDUPmJcA6l0Sl4A6nyZCspDSWh8UO17wEcW Delete

ghp\_ICDUPmJcA6l0Sl4A6nyZCspDSWh8UO17wEcW

**6º** Una vez que tenemos el token vamos con la creación de repositorio. Para crear el repositorio nos vamos a repositories y damos a New.





**7º** en el próximo paso vamos a darle nombre, visibilidad y otros detalles al repositorio. Cuando ya tenemos todo damos a create repository.

### Create a new repository


A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)


**Owner \*** **Repository name \***

 EvaPul / Actividad 

Great repository names are short and memorable. Need inspiration? How about [fictional-guide?](#)

**Description (optional)**

☒  **Public**  
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**  
You choose who can see and commit to this repository.

**Initialize this repository with:**  
Skip this step if you're importing an existing repository.


☒ **Add a README file**  
This is where you can write a long description for your project. [Learn more.](#)


**Add .gitignore**  
Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: **None**

**Choose a license**  
A license tells others what they can and can't do with your code. [Learn more.](#)

License: **None**

This will set  **main** as the default branch. Change the default name in your [settings](#).

 You are creating a public repository in your personal account.

**Create repository**

**8º** Abrimos el terminal con Ctrl+Alt+T y empezamos con los comandos. Primero entramos en la carpeta anteriormente creada llamada prueba donde hemos ido trabajando.

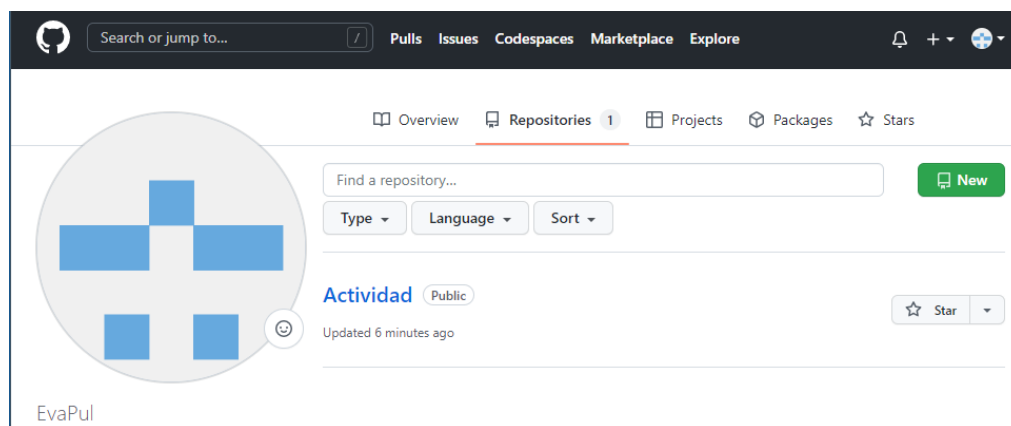
Comenzamos con el comando `git remote add prueba <Url del repositorio>` para poder subir los archivos al repositorio.

```
eva@eva-VirtualBox:~/Escritorio/prueba$ git remote add prueba https://github.com/EvaPul/Actividad.git
```

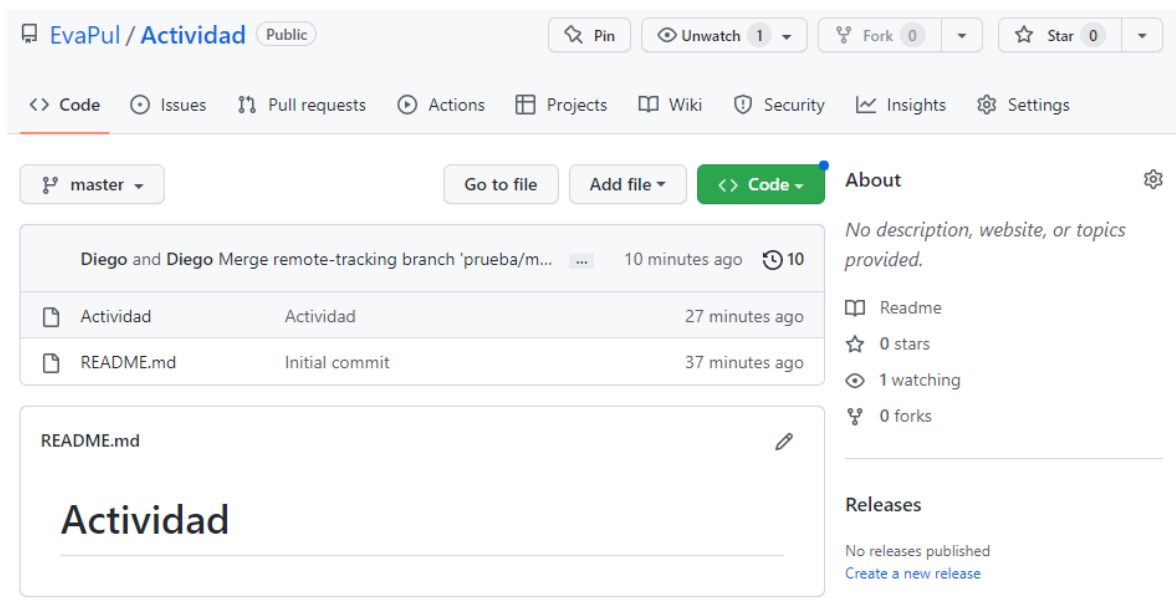
9º Utilizamos el comando `git push -u prueba master` para subir los archivos al repositorio indicado con el comando anterior.

```
eva@eva-VirtualBox:~/Escritorio/prueba$ git push -u prueba master
Username for 'https://github.com': ghp_ICDUPmJcA6l0Sl4A6nyZCspDSWh8U017wEcW
Password for 'https://ghp_ICDUPmJcA6l0Sl4A6nyZCspDSWh8U017wEcW@github.com':
Enumerando objetos: 9, listo.
Contando objetos: 100% (9/9), listo.
Compresión delta usando hasta 4 hilos
Comprimiendo objetos: 100% (6/6), listo.
Escribiendo objetos: 100% (9/9), 806 bytes | 806.00 KiB/s, listo.
Total 9 (delta 1), reusado 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:   https://github.com/EvaPul/Actividad/pull/new/master
remote:
To https://github.com/EvaPul/Actividad.git
 * [new branch]      master -> master
Rama 'master' configurada para hacer seguimiento a la rama remota 'master' de 'prueba'.
```

10º Si vamos a GitHub a repositorios podemos ver que ha tenido una actualización hace tan solo 6 minutos. Si pinchamos en Actividad podemos ver el repositorio.



11º Dentro podemos ver las actualizaciones realizadas en el repositorio.



**12º** Vamos a la terminal del compañero y ejecutamos el comando `git clone <Url del repositorio>` para descargar el repositorio en la máquina del compañero y poder trabajar en él.

```
diego@diego-VirtualBox:~/Prueba$ git clone https://github.com/EvaPul/Actividad.git
Clonando en 'Actividad'...
remote: Enumerating objects: 12, done.
remote: Counting objects: 100% (12/12), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 12 (delta 1), reused 9 (delta 1), pack-reused 0
Desempaquetando objetos: 100% (12/12), 1.33 KiB | 681.00 KiB/s, listo.
```

**13º** Una vez que hemos descargado el repositorio ejecutamos el comando `touch <nombre fichero>` para crear el fichero, en este caso `Actividad`.

Después ejecutamos en comando `git add <nombre fichero>` para añadir stage. Una vez que lo tenemos en el stage ejecutamos el comando `nano <nombre fichero>` para editar el archivo.

Una vez editado el archivo volvemos a ejecutar el comando `git add <nombre fichero>` para actualizar el fichero que tenemos en stage.

Finalmente ejecutamos el comando `git commit -m "<nombre que se le quiera dar para identificarlo>"` para generar un commit.

```
diego@diego-VirtualBox:~/Prueba/Actividad$ touch Actividad
diego@diego-VirtualBox:~/Prueba/Actividad$ git add Actividad
diego@diego-VirtualBox:~/Prueba/Actividad$ nano Actividad
diego@diego-VirtualBox:~/Prueba/Actividad$ git add Actividad
diego@diego-VirtualBox:~/Prueba/Actividad$ git commit -m "Actividad"
[main 9d8774b] Actividad
1 file changed, 1 insertion(+)
create mode 100644 Actividad
```

**14º** una vez creado el commit ejecutamos el comando `git push prueba master --force` para subir de forma forzada el commit al repositorio porque la página puede crear un main (como ha pasado en este caso).

```
diego@diego-VirtualBox:~/Prueba$ git push prueba master --force
Username for 'https://github.com': ghp_ICDUPmJcA6l0Sl4A6nyZCspDSWh8U017wEcW
Password for 'https://ghp_ICDUPmJcA6l0Sl4A6nyZCspDSWh8U017wEcW@github.com':
Enumerando objetos: 23, listo.
Contando objetos: 100% (23/23), listo.
Compresión delta usando hasta 4 hilos
Comprimiendo objetos: 100% (17/17), listo.
Escribiendo objetos: 100% (23/23), 2.10 KiB | 2.10 MiB/s, listo.
Total 23 (delta 1), reusado 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/EvaPul/Actividad.git
+ b0868da...f216838 master -> master (forced update)
```

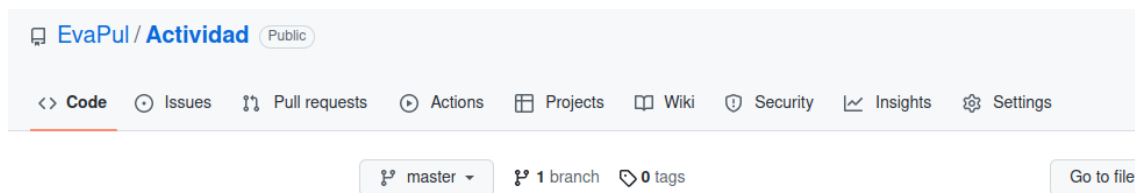
**15º** realizamos el comando `git merge Prueba/main --allow-unrelated-histories` para poder fusionar el branch main y master.

```
diego@diego-VirtualBox:~/Prueba$ git merge prueba/main --allow-unrelated-histories
Merge made by the 'recursive' strategy.
README.md | 1 +
1 file changed, 1 insertion(+)
create mode 100644 README.md
```

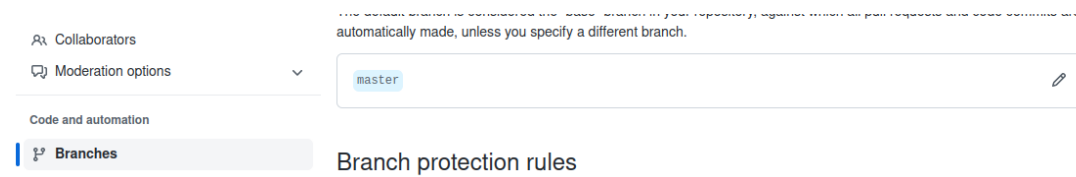
**16°** Una vez que hemos fusionado el main y master ejecutamos el comando `git push prueba master --force` para poder subir el repositorio.

```
diego@diego-VirtualBox:~/Prueba$ git push prueba master --force
Username for 'https://github.com': ghp_ICDUPmJcA6l0Sl4A6nyZCspDSWh8U017wEcW
Password for 'https://ghp_ICDUPmJcA6l0Sl4A6nyZCspDSWh8U017wEcW@github.com':
Enumerando objetos: 4, listo.
Contando objetos: 100% (4/4), listo.
Compresión delta usando hasta 4 hilos
Comprimiendo objetos: 100% (2/2), listo.
Escribiendo objetos: 100% (2/2), 324 bytes | 324.00 KiB/s, listo.
Total 2 (delta 0), reusado 0 (delta 0)
To https://github.com/EvaPul/Actividad.git
f216838..4bcc0e0 master -> master
```

**17°** Hemos tenido que cambiar la Branch principal y para ello hemos ido al repositorio y hemos pinchado sobre settings.



**18°** Una dentro de Settings podemos ver que nos aparece la opción Branches y pinchamos ahí. Automáticamente a la derecha nos aparece un recuadro donde podremos cambiar la Branch.



**19°** Una vez que hemos cambiado la Branch nos vamos al terminal (continuamos en la del compañero) y ejecutamos el comando `git push prueba --delete main` para eliminar main.

```
diego@diego-VirtualBox:~/Prueba$ git push prueba --delete main
Username for 'https://github.com': ghp_ICDUPmJcA6l0Sl4A6nyZCspDSWh8U017wEcW
Password for 'https://ghp_ICDUPmJcA6l0Sl4A6nyZCspDSWh8U017wEcW@github.com':
To https://github.com/EvaPul/Actividad.git
- [deleted]          main
```

**20°** Ahora nos volvemos a la terminal del compañero inicial y ejecutamos el comando `git clone <Url repositorio>` para descargar en la maquina el repositorio actualizado por el otro compañero.

En este caso se realizó el comando `git clone` en vez de `git pull` porque estaba dando una serie de problemas de sincronización de archivos.

```
eva@eva-VirtualBox:~/Escritorio/prueba$ git clone https://github.com/EvaPul/Actividad.git
Clonando en 'Actividad'...
remote: Enumerating objects: 28, done.
remote: Counting objects: 100% (28/28), done.
remote: Compressing objects: 100% (19/19), done.
remote: Total 28 (delta 1), reused 25 (delta 1), pack-reused 0
Desempaquetando objetos: 100% (28/28), 2.93 KiB | 600.00 KiB/s, listo.
```

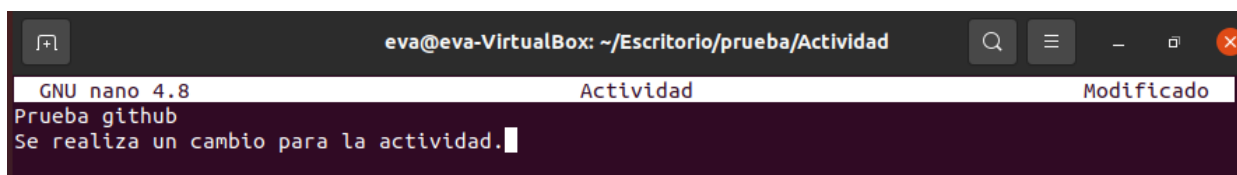
**22º** Ahora que ya tenemos el repositorio actualizado descargado en la maquina vamos a realizar una nueva modificación.

Para realizar la modificación nos situamos en prueba y ejecutamos el comando ls para listar y así poder ver los ficheros que tenemos en prueba. Podemos ver que tenemos el fichero Actividad y ejecutamos el comando cd Actividad para entrar en Actividad. Volvemos a realizar el comando ls para poder ver los ficheros que se encuentran dentro de Actividad y como vemos que tenemos el fichero Actividad que ha creado el compañero y ha modificado vamos a realizarle una nueva modificación.

Ejecutamos el comando nano <nombre del fichero> para abrirlo y modificarlo.

```
eva@eva-VirtualBox:~/Escritorio/prueba$ ls
Actividad aprende-git nuevodoc.txt prueba.txt
eva@eva-VirtualBox:~/Escritorio/prueba$ cd Actividad
eva@eva-VirtualBox:~/Escritorio/prueba/Actividad$ ls
Actividad README.md
eva@eva-VirtualBox:~/Escritorio/prueba/Actividad$ nano Actividad
```

**23º** Una vez abierto el fichero podemos ver la frase escrita por el compañero "Prueba github" y nosotros vamos a añadir una segunda frase "Se realiza un cambio para la actividad". Una vez realizado el cambio pulsamos Ctrl+O para guardar el cambio y cerrar el archivo.



```
eva@eva-VirtualBox: ~/Escritorio/prueba/Actividad
GNU nano 4.8 Actividad Modificado
Prueba github
Se realiza un cambio para la actividad.
```

**24º** Una vez editado el archivo volvemos ejecutamos el comando git add <nombre fichero> para añadirlo al stage.

```
eva@eva-VirtualBox:~/Escritorio/prueba/Actividad$ git add Actividad
```

**25º** Ahora ya ejecutamos el comando git commit -m "<nombre que se le quiera dar para identificarlo>" para generar un commit.

```
eva@eva-VirtualBox:~/Escritorio/prueba/Actividad$ git commit -m "Actividad"
[master e064ecb] Actividad
1 file changed, 1 insertion(+)
```

**26º** Realizamos el comando cd .. para salir de Actividad e ir a prueba.

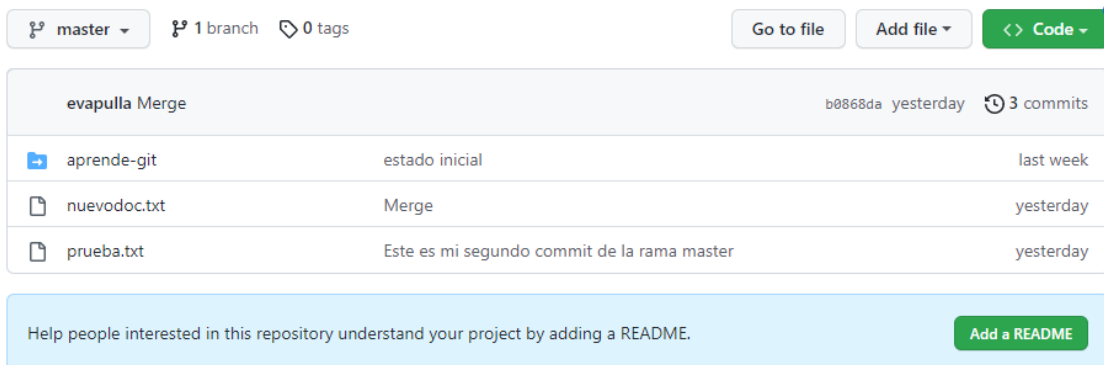
```
eva@eva-VirtualBox:~/Escritorio/prueba/Actividad$ cd ..
```



**27º** Ahora ya desde prueba realizamos el comando git push prueba master --force -force para poder subir el repositorio ya editado.

```
eva@eva-VirtualBox:~/Escritorio/prueba$ git push prueba master --force
Username for 'https://github.com': ghp_ICDUPmJcA6l0Sl4A6nyZCspDSWh8U017wEcW
Password for 'https://ghp_ICDUPmJcA6l0Sl4A6nyZCspDSWh8U017wEcW@github.com':
Enumerando objetos: 9, listo.
Contando objetos: 100% (9/9), listo.
Compresión delta usando hasta 4 hilos
Comprimiendo objetos: 100% (6/6), listo.
Escribiendo objetos: 100% (9/9), 806 bytes | 806.00 KiB/s, listo.
Total 9 (delta 1), reusado 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/EvaPul/Actividad.git
+ 4bcc0e0...b0868da master -> master (forced update)
```

**28º** Si vamos a GitHub podemos ver las actualizaciones en el repositorio.



evapulla Merge b0868da yesterday 3 commits

File	Commit Message	Time
aprende-git	estado inicial	last week
nuevodoc.txt	Merge	yesterday
prueba.txt	Este es mi segundo commit de la rama master	yesterday

Help people interested in this repository understand your project by adding a README. [Add a README](#)

**28º** En el siguiente paso lo que vamos a hacer es mover los ficheros que tenemos en Actividad a prueba y para ello realizamos el comando mv Actividad ../

```
eva@eva-VirtualBox:~/Escritorio/prueba$ ls
Actividad aprende-git nuevodoc.txt prueba.txt
eva@eva-VirtualBox:~/Escritorio/prueba$ cd Actividad
eva@eva-VirtualBox:~/Escritorio/prueba/Actividad$ ls
Actividad README.md
eva@eva-VirtualBox:~/Escritorio/prueba/Actividad$ mv Actividad ../
```

**29º** En nuestro caso tuvimos problemas de permisos en la carpeta y lo solucionamos con el comando sudo chmod 777 -R prueba para poder dar permisos de todo tanto al directorio como a los ficheros.

```
eva@eva-VirtualBox:~/Escritorio$ sudo chmod 777 -R prueba
```

**30º** Luego ejecutamos el comando git add \* para agregar todos los archivos al stage.

```
eva@eva-VirtualBox:~/Escritorio/prueba$ git add *
```

**31º** una vez que ya está subido al stage realizamos el comando `git status -s` para comprobar que se ha subido correctamente.

```
eva@eva-VirtualBox:~/Escritorio/prueba$ git add *
eva@eva-VirtualBox:~/Escritorio/prueba$ git status -s
A   Actividad
A   README.md
M   aprende-git
M   nuevodoc.txt
M   prueba.txt
```

**32º** Ejecutamos el comando `git commit -m "<nombre que se le quiera dar para identificarlo>"` para generar un commit.

```
eva@eva-VirtualBox:~/Escritorio/prueba$ git commit -m "Actividad"
[master 8b06544] Actividad
4 files changed, 3 insertions(+)
create mode 100755 Actividad
create mode 100755 README.md
mode change 100644 => 100755 nuevodoc.txt
mode change 100644 => 100755 prueba.txt
```

**33º** Finalmente ejecutamos el comando `git push prueba master --force` para poder subir el repositorio.

```
eva@eva-VirtualBox:~/Escritorio/prueba$ git push -u prueba master
Username for 'https://github.com': ghp_ICDUPmJcA6l0Sl4A6nyZCspDSWh8U017wEcW
Password for 'https://ghp_ICDUPmJcA6l0Sl4A6nyZCspDSWh8U017wEcW@github.com':
Enumerando objetos: 5, listo.
Contando objetos: 100% (5/5), listo.
Compresión delta usando hasta 4 hilos
Comprimiendo objetos: 100% (3/3), listo.
Escribiendo objetos: 100% (4/4), 457 bytes | 457.00 KiB/s, listo.
Total 4 (delta 0), reusado 0 (delta 0)
To https://github.com/EvaPul/Actividad.git
 b0868da..8b06544  master -> master
Rama 'master' configurada para hacer seguimiento a la rama remota 'master' de 'prueba'.
```

**34º** Si vamos a nuestro repositorio en GitHub podemos visualizar las actualizaciones del repositorio.

master
1 branch
0 tags
Go to file
Add file
Code

evapulla Actividad		8b06544 14 minutes ago	4 commits
aprende-git	estado inicial		last week
Actividad	Actividad		14 minutes ago
README.md	Actividad		14 minutes ago
nuevodoc.txt	Actividad		14 minutes ago
prueba.txt	Actividad		14 minutes ago

README.md

## Actividad

**35°** Para el siguiente paso nos vamos al terminal del otro compañero, en este caso Diego, para realizar unos nuevos cambios (meter una línea más de texto).

Una vez que hemos realizado los cambios ejecutamos el comando `git add <nombre archivo>` para subirlo al stage y finalmente realizamos el comando `git commit -m "<nombre que se le quiera dar para identificarlo>"`.

```
diego@diego-VirtualBox:~/Prueba$ nano nuevodoc.txt
diego@diego-VirtualBox:~/Prueba$ git add nuevodoc.txt
diego@diego-VirtualBox:~/Prueba$ git commit -m "Actividad"
[master c0165de] Actividad
1 file changed, 2 insertions(+)
create mode 100755 nuevodoc.txt
```

**36°** Una vez realizado el commit ejecutamos el comando `git push prueba master` para volver a subirlo al repositorio.

```
diego@diego-VirtualBox:~/Prueba$ git push prueba master
Username for 'https://github.com': ghp_ICDUPmJcA6l0Sl4A6nyZCspDSWh8U017wEcW
Password for 'https://ghp_ICDUPmJcA6l0Sl4A6nyZCspDSWh8U017wEcW@github.com':
Enumerando objetos: 48, listo.
Contando objetos: 100% (44/44), listo.
Compresión delta usando hasta 4 hilos
Comprimiendo objetos: 100% (29/29), listo.
Escribiendo objetos: 100% (37/37), 4.11 KiB | 4.11 MiB/s, listo.
Total 37 (delta 5), reusado 0 (delta 0)
```


**37°** Vamos al terminal del otro compañero, en este caso Eva y realizamos el comando `git pull prueba master` para bajarnos las modificaciones del repositorio.


```
eva@eva-VirtualBox:~/Escritorio/prueba$ git pull prueba master
remote: Enumerating objects: 48, done.
remote: Counting objects: 100% (44/44), done.
remote: Compressing objects: 100% (24/24), done.
remote: Total 37 (delta 6), reused 36 (delta 5), pack-reused 0
Desempaquetando objetos: 100% (37/37), 4.04 KiB | 1.35 MiB/s, listo.
Desde https://github.com/EvaPul/Actividad
* branch                master      -> FETCH_HEAD
  7c9fa88..c0165de master      -> prueba/master
Actualizando 7c9fa88..c0165de
warning: unable to rmdir 'aprende-git': El directorio no está vacío
Fast-forward
 aprende-git | 1 -
 nuevodoc.txt | 1 +
 prueba.txt   | 1 -
3 files changed, 1 insertion(+), 2 deletions(-)
delete mode 160000 aprende-git
delete mode 100755 prueba.txt
```

**38°** En GitHub vemos finalmente como quedaría nuestro repositorio.

master ▾
Go to file
Add file ▾
<> Code ▾

Diego and Diego Actividad ...		4 minutes ago ⌚ 20
📄 Actividad	Actividad	8 minutes ago
📄 README.md	Actividad	47 minutes ago
📄 nuevodoc.txt	Actividad	4 minutes ago

README.md 



# Actividad

---