

# Trabajo 1 - Introducción a Git

Por: **Adrián Pisabarro García**

Puedes tener una mejor vista en: <https://github.com/adrianpisabarrogarcia/daw-froga-git/blob/master/git-trabajo/GUION.md>

## 1. Iniciar nuestro Repositorio desde local

1.- Lo primero es situarnos en la carpeta donde vamos a crear nuestro repositorio git en mi caso:

```
$ cd Desktop/Git/ .
```

Ahora creamos la carpeta: `$ mkdir daw-froga-git` .

Y accedemos a ella: `$ cd daw-froga-git/`

2.- Utilizamos el comando Git, ahora sí para inicializar nuestro repo en local: `$ git init`

3.- Vamos a añadir el archivo *README.md* que nos pide, para eso lo un `$ nano README.md` creamos el archivo Markdown.

4.- Añadimos los archivos a la cola todos los archivos con un `$ git add *` , para comprobar el estado podemos incluso poner un `$ git status` . Ahora un commit para que quede reflejada una versión que subiremos con `$ git commit -m 'add README.md'` y añadimos un mensaje que indique claro y conciso el cambio que hemos hecho. Debemos fijarnos en que hace la inserción.

5.- En estos momentos vamos a sincronizar nuestro repo en local con el repo en GitHub, para ello antes tendremos que crearlo en [GitHub](#) con el mismo nombre previamente y en la terminal añadimos el siguiente comando para enlazarlo:

```
$ git remote set-url origin https://github.com/adrianpisabarrogarcia/daw-froga-git.git .
```

6.- Ya podemos hacer un *push* para subir todo al repo online: `git push -u origin master` y subirlo a nuestro *origin* en la rama llamada *master*.

## Create a new repository

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

### Repository template

Start your repository with a template repository's contents.

No template ▾

Owner \*



adrianpisabarrogarcia ▾

Repository name \*

daw-froga-git ✓

Great repository names are short and memorable. Need inspiration? How about **fictional-sniffle**?

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](#).

☐ **Choose a license**

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

Create repository

## 2. Crea una rama con otra versión del proyecto

1.- Crear el archivo **adrian.html** y subirlo a la rama master.

```
$ nano adrian.html
```

 Creación del archivo **adrian.html**.

```
$ git add *
```

 ó 

```
$ git add adrian.html
```

 Añadir a la cola de subida el archivo **adrian.html**.

```
$ git commit -m "add file adrian.html to master branch"
```

 Hacer un control de versión.

```
$ git push origin master
```

 Subir de local a remoto.

```
~/D/G/daw-froga-git
nothing added to commit but untracked files present (use "git add" to track)
adrianpisabarrogarcia ~/D/G/daw-froga-git git add * Tue Sep 22 07:33:30 2020
adrianpisabarrogarcia ~/D/G/daw-froga-git git status Tue Sep 22 07:33:35 2020
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   adrian.html

adrianpisabarrogarcia ~/D/G/daw-froga-git git commit -m "add file adrian.html to master brach"
[master 6a5765f] add file adrian.html to master brach
1 file changed, 9 insertions(+)
create mode 100644 adrian.html
adrianpisabarrogarcia ~/D/G/daw-froga-git git push -u origin master Tue Sep 22 07:33:42 2020
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 394 bytes | 394.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/adrianpisabarrogarcia/daw-froga-git.git
   acff3d5..6a5765f  master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
adrianpisabarrogarcia ~/D/G/daw-froga-git 3991ms < Tue Sep 22 07:33:56 2020
```



adrianpisabarrogarcia add file adrian.html to master brach

6a5765f 21 seco



README.md

add README.md



adrian.html

add file adrian.html to master brach

README.md

Esto es una prueba

2.- Añadimos una rama: `$ git branch test`.

3.- Movernos a la rama auxiliar: `$ git checkout test`. Saldrá esto: `Switched to branch 'test'`.

4.- Modifica el archivo **adrian.html** y súbelo a la segunda rama.

`$ git nano adrian.html` Hacemos todos los cambios que tenemos que hacer al archivo.

`$ git add *` ó `$ git add adrian.html` Añadir a la cola de subida el archivo **adrian.html**.

`$ git commit -m "v2 file: adrian.html"` Hacer un control de versión.

`$ git push origin master` Subir de local a remoto.

5.- Comprobación de las dos ramas con los dos archivos diferenciados:

```
~/D/G/daw-froga-git
Branch 'master' set up to track remote branch 'master' from 'origin'.
Everything up-to-date
adrianpisabarrogarcia ~/D/G/daw-froga-git test git add * 1115ms < Tue Sep 22 07:45:46 2020
adrianpisabarrogarcia ~/D/G/daw-froga-git test git status Tue Sep 22 07:46:18 2020
On branch test
nothing to commit, working tree clean
adrianpisabarrogarcia ~/D/G/daw-froga-git test git commit -m "v2 file: adrian.html"
On branch test
nothing to commit, working tree clean
! adrianpisabarrogarcia ~/D/G/daw-froga-git test git push -u origin test
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 359 bytes | 359.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote:
remote: Create a pull request for 'test' on GitHub by visiting:
remote: https://github.com/adrianpisabarrogarcia/daw-froga-git/pull/new/test
remote:
To https://github.com/adrianpisabarrogarcia/daw-froga-git.git
* [new branch] test -> test
Branch 'test' set up to track remote branch 'test' from 'origin'.
adrianpisabarrogarcia ~/D/G/daw-froga-git test 4263ms < Tue Sep 22 07:47:10 2020
```

/ daw-froga-git

Pull requests Actions Projects Wiki Security Insights Settings

test had recent pushes less than a minute ago

Compare & pull request

test 2 branches 0 tags




Go to file

Add file

Code

This branch is 1 commit ahead of master.

Pull request Compare

	adrianpisabarrogarcia v2 file: adrian.html	492233d 2 minutes ago	3 commits
	README.md	add README.md	40 minutes ago
	adrian.html	v2 file: adrian.html	2 minutes ago

- ¿Cómo veo las ramas que existen? `$ git branch`.
- ¿Cómo sé cuál es la rama activa? Se selecciona con un `* test`.

### 3. Une la rama del ejercicio2 a la rama master.

- 1.- Primero nos vamos a situar en la rama destino: `$ git checkout master`.
- 2.- Después de añadir cambios vamos a unir las dos ramas, la rama **test** en la **master**:  
`$ git merge test`.
- 3.- Por último se añade un `$ git push` para ver en remoto que he hecho los cambios.

The image shows a terminal window with the following commands and output:

```
To https://github.com/adrianpisabarrogarcia/daw-froga-git.git
* [new branch]      test -> test
Branch 'test' set up to track remote branch 'test' from 'origin'.
adrianpisabarrogarcia ~/D/G/daw-froga-git > test git branch 4263ms < Tue Sep 22 07:47:10 2020
git: 'brach' is not a git command. See 'git --help'.

The most similar command is
branch
! adrianpisabarrogarcia ~/D/G/daw-froga-git > test git branch master Tue Sep 22 07:53:01 2020
* test
adrianpisabarrogarcia ~/D/G/daw-froga-git > test git checkout master Tue Sep 22 08:08:42 2020
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
adrianpisabarrogarcia ~/D/G/daw-froga-git > git merge test Tue Sep 22 08:08:48 2020
Updating 6a5765f..492233d
Fast-forward
 adrian.html | 4 ++--
 1 file changed, 2 insertions(+), 2 deletions(-)
adrianpisabarrogarcia ~/D/G/daw-froga-git > git push Tue Sep 22 08:08:49 2020
Total 0 (delta 0), reused 0 (delta 0)
To https://github.com/adrianpisabarrogarcia/daw-froga-git.git
 6a5765f..492233d master -> master
adrianpisabarrogarcia ~/D/G/daw-froga-git >
```

Below the terminal, the GitHub repository page for `daw-froga-git` is visible, showing the master branch, 2 branches, 0 tags, and a list of commits.

## 4. Volver hacia atrás en los commits

### Opción 1: GIT RESET

1.- Primero vamos a localizar el id del cambio con: `$ git log` yo voy a escoger este id: `492233ddcae0faaa063a5b71792b7d991a8113e7` ó `492233d`. Podemos utilizar para no tener tanto texto la opción de `$ git log --oneline | cat`. En este primer paso también podemos hacer uso del `git reset --hard HEAD~n` donde n indica el numero de commit anteriores. Por cierto si incluimos el `--hard` no haría falta hacer un add o un commit, si no lo incluimos sí.

```
492233d v2 file: adrian.html
6a5765f add file adrian.html to master branch
acff3d5 add README.md
```

2.- Hacemos un reset del commit específico:

`$ git reset 492233ddcae0faaa063a5b71792b7d991a8113e7` en este momento nuestro cambio estará en el stage, habrá que hacer un add, un commit, un merge y posteriormente un push por no haber añadido el `--hard`.

### Opción 2: GIT CHECKOUT

1.- `$ git log` para saber cual va a ser nuestro id en este caso será:

```
b691cfe951644e2d88c651038dcea29c188075bb .
```

2.- Ahora nos situaremos en la rama que estara nuestro puntero HEAD:

```
$ git checkout b691cfe951644e2d88c651038dcea29c188075bb .
```

3.- En estos momentos nuestro repo ya habrá sufrido los cambios al haber puesto el punto, faltaría subir los cambios.

### Opción 3: GIT REVERT

Parecido al `$ git reset` pero compara archivos como si hiciera un

```
$ git diff archivo1 archivo2 .
```

### ¿Qué diferencia hay?

- Cuando utilizamos el `$ git reset --hard HEAD~n` lo que hace es identificar cual es el commit n numeros atrás y añadimos una nueva rama que la tendremos que poner en nuestra rama final. Mueves un puntero llamado *HEAD* a un momento específico.
- Cuando utilizamos el `$ git checkout id-log .` directamente identificamos y marcamos a que versión se va a revertir.

## 5. Crear una subrama

Vamos a imitar esta imagen solo que cambiando la llamada "Experimental" por "test" y "otra rama" por "sub-test":



Creamos las dos ramas gracias a `$ git branch test` y `$ git branch sub-test` para hacer "sub-test" hay que colocarse en "test". Además voy a editar el archivo *adrian.html* en "test" y también en "sub-test" para unirlos.

```
$ git branch test
$ git checkout test
$ nano adrian.html (-)




$ git branch sub-test
$ git checkout sub-test
$ nano adrian.html (-)



$ git checkout test
$ git merge sub-test



$ git checkout master
$ git merge test (-)
```

(-) con sus correspondientes add y commit.

```
~ /D/G/daw-froga-git
README.md  adrian.html
adrianpisabarrogarcia ~ /D/G/daw-froga-git > ↵ sub-test nano adrian.html
Tue Sep 22 19:34:47 2020
adrianpisabarrogarcia ~ /D/G/daw-froga-git > ↵ sub-test * git add *
9.1s < Tue Sep 22 19:36:22 2020
adrianpisabarrogarcia ~ /D/G/daw-froga-git > ↵ sub-test ~ git commit -m "de sub-tree a tree"
Tue Sep 22 19:36:27 2020
[sub-test a602e68] de sub-tree a tree
1 file changed, 2 insertions(+), 2 deletions(-)
adrianpisabarrogarcia ~ /D/G/daw-froga-git > ↵ sub-test git checkout test
Tue Sep 22 19:39:14 2020
Switched to branch 'test'
Your branch is up to date with 'origin/test'.
adrianpisabarrogarcia ~ /D/G/daw-froga-git > ↵ test git merge sub-test
Tue Sep 22 19:40:13 2020
Updating 492233d..a602e68
Fast-forward
 adrian.html | 4 ++--
1 file changed, 2 insertions(+), 2 deletions(-)
adrianpisabarrogarcia ~ /D/G/daw-froga-git > ↵ test + git checkout master
Tue Sep 22 19:40:18 2020
Switched to branch 'master'
Your branch and 'origin/master' have diverged,
and have 1 and 3 different commits each, respectively.
(use "git pull" to merge the remote branch into yours)
adrianpisabarrogarcia ~ /D/G/daw-froga-git > ↵ ± git merge test
Tue Sep 22 19:40:30 2020
Auto-merging adrian.html
CONFLICT (content): Merge conflict in adrian.html
Automatic merge failed; fix conflicts and then commit the result.
! adrianpisabarrogarcia ~ /D/G/daw-froga-git > ↵ *~± git add *
Tue Sep 22 19:40:42 2020
adrianpisabarrogarcia ~ /D/G/daw-froga-git > ↵ ~± git add *
Tue Sep 22 19:40:46 2020
adrianpisabarrogarcia ~ /D/G/daw-froga-git > ↵ ~± git commit -m "último merge de test a master"
Tue Sep 22 19:42:26 2020
[master f28d063] último merge de test a master
adrianpisabarrogarcia ~ /D/G/daw-froga-git > ↵ ± git push origin master
```

 master ▾  2 branches  0 tags Go to file Add file ▾ Code ▾

 **adrianpisabarrogarcia** último merge de test a master f28d063 32 seconds ago  19 commits

 README.md	add README.md	13 hours ago
 adrian.html	último merge de test a master	32 seconds ago

## 6. Importar proyecto a escritorio

- 1.- Nos situamos en el escritorio con la terminal: `$ cd $HOME/Desktop` .
- 2.- Hacemos un *clone* con el enlace de GitHub:  
`$ git clone https://github.com/adrianpisabarrogarcia` .
- 3.- Accedemos con un `$ cd <nombre-carpeta>` .

```
adrianpisabarrogarcia ~/D/G/daw-froga-git > cd $HOME/Desktop/
adrianpisabarrogarcia ~/Desktop > git clone https://github.com/adrianpisabarrogarcia/daw-froga-git
Cloning into 'daw-froga-git'...
remote: Enumerating objects: 45, done.
remote: Counting objects: 100% (45/45), done.
remote: Compressing objects: 100% (36/36), done.
remote: Total 45 (delta 12), reused 40 (delta 7), pack-reused 0
Unpacking objects: 100% (45/45), done.
adrianpisabarrogarcia ~/Desktop > ls
RPJ-Newsletter plantilla septiembre.docx
adrianpisabarrogarcia ~/Desktop > cd daw-froga-git/
adrianpisabarrogarcia ~/D/daw-froga-git > ls
README.md  adrian.html
adrianpisabarrogarcia ~/D/daw-froga-git >
```

## 7.- Añade de colaborador y que edite la otra persona

- 1.- Me he creado otra cuenta de GitHub con otro correo electrónico.
- 2.- Desde mi cuenta habitual lo que he hecho ha sido ir *settings* y después a *Manage access*.
- 3.- El otro usuario llamado *pisiadriantxu* ha hecho un cambio en el README.md y lo ha commiteado.

Imagen 1:

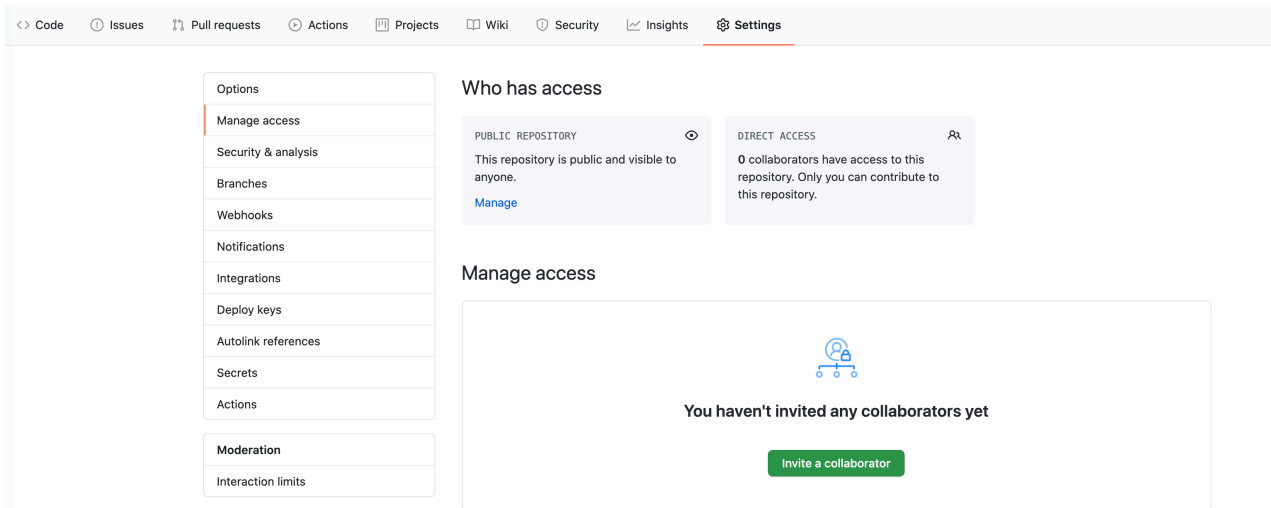


Imagen 2:



adrianpisabarrogarciadaw-froga-git

Watch1

Star0

Fork0

<>Code

Issues

Pull requests

Actions

Projects

Wiki

Security

Insights

master

daw-froga-git / README.md

Go to file

pisiadriantxu

Update pisiadriantxu README.md

Latest commit 4dbdf3d now

History

2 contributors

2 lines (2 sloc)84 Bytes

RawBlame

Esto es una prueba

El segundo editor (pisiadriantxu) ha cambiado este archivo