

Git & GitHub



Índice Sesión 4



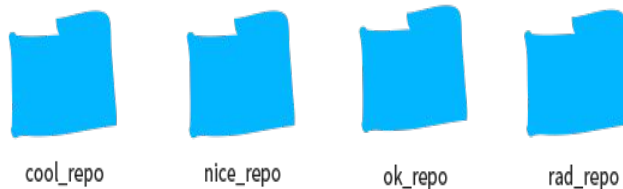
- 1. Seguimiento de un repositorio local**
- 2. Otros Comandos de Git**

Git Repositorios: Seguimiento

Un **repositorio** es una colección de elementos relacionados. En nuestro caso, cuando estamos escribiendo software, es una colección de archivos relacionados a un proyecto de software. Puedes imaginarlo como la carpeta de un proyecto con todos los archivos relevantes en su interior. De hecho, así es como se verá en tu ordenador de todas maneras. Algunas veces se les llama también "repos" como abreviación.

Le dices a Git qué carpeta contiene tu proyecto y Git empezará a hacer el seguimiento de todos los cambios en esa carpeta. Esto es lo que conforma un repositorio de Git: una carpeta de elementos que son seguidos por Git. Git hace seguimiento de cuándo se añaden archivos, cuándo se extraen e incluso cuándo una simple letra en un simple archivo ha sido cambiada. A todo esto Git le añade quién y cuándo lo hizo. En software, al proceso de hacer seguimiento a los cambios se le llama **control de versiones**.

MY REPOS FOLDER



Git Repositorios: Seguimiento

Dile a Git que inicie (empiece el seguimiento) de la carpeta en la que estás actualmente:

```
git init
```

El último comando debe devolver algo como "Initialized empty Git repository". Los otros comandos no tienen retorno, es decir, no devuelven ningún mensaje.

¡Lo lograste! Si quieres asegurarte de que es un repositorio de Git, escribe git status y si no devuelve 'fatal: Not a git repository...', ¡está hecho!

Para añadir todos los archivos con cambios

```
git add .
```

```
git commit -m "Comentario"
```

Git Repositorios: Seguimiento

Si quieres sincronizar tu versión **local** con la que está guardada GitHub.com, primero crea un nuevo repositorio **remoto (remote)** en GitHub.com. Vuelve a la terminal, y situado dentro de la carpeta 'hola-mundo' que inicializaste como repositorio de Git en el reto anterior, vas a decirle a Git que recuerde la dirección de la versión remota en los servidores GitHub.

- Ve a github.com, haz login, y haz clic en el '+' en la parte derecha arriba y después en 'Nuevo repositorio' ('New repository').
- Dale un nombre que coincida con el nombre de tu repositorio local, y una descripción corta.
- Hazlo público. Esto significa que será listado en tu perfil público.
- No inicialices con un archivo README porque ya tienes un archivo,

Vuelve a la terminal, y situado dentro de la carpeta 'hola-mundo' que inicializaste como repositorio de Git en el reto anterior, vas a decirle a Git que recuerde la dirección de la versión remota en los servidores GitHub.

```
git remote add origin <URLDEGITHUB>  
git push origin master
```

git log



```
$ git log
commit ca82a6dffa817ec66f44342087202690a93763949
Author: Scott Chacon <schacon@gee-mail.com>
Date:   Mon Mar 17 21:52:11 2008 -0700

    changed the version number

commit 085bb3bcb608e1e8451d4b2432f8ecbe6306e7e7
Author: Scott Chacon <schacon@gee-mail.com>
Date:   Sat Mar 15 16:40:33 2008 -0700

    removed unnecessary test

commit a11bef06a3f659402fe7563abf99ad00de2209e6
Author: Scott Chacon <schacon@gee-mail.com>
Date:   Sat Mar 15 10:31:28 2008 -0700

    first commit
```

git tag



```
$ git log
commit ca82a6dffa817ec66f44342087202690a93763949
Author: Scott Chacon <schacon@gee-mail.com>
Date:   Mon Mar 17 21:52:11 2008 -0700

    changed the version number

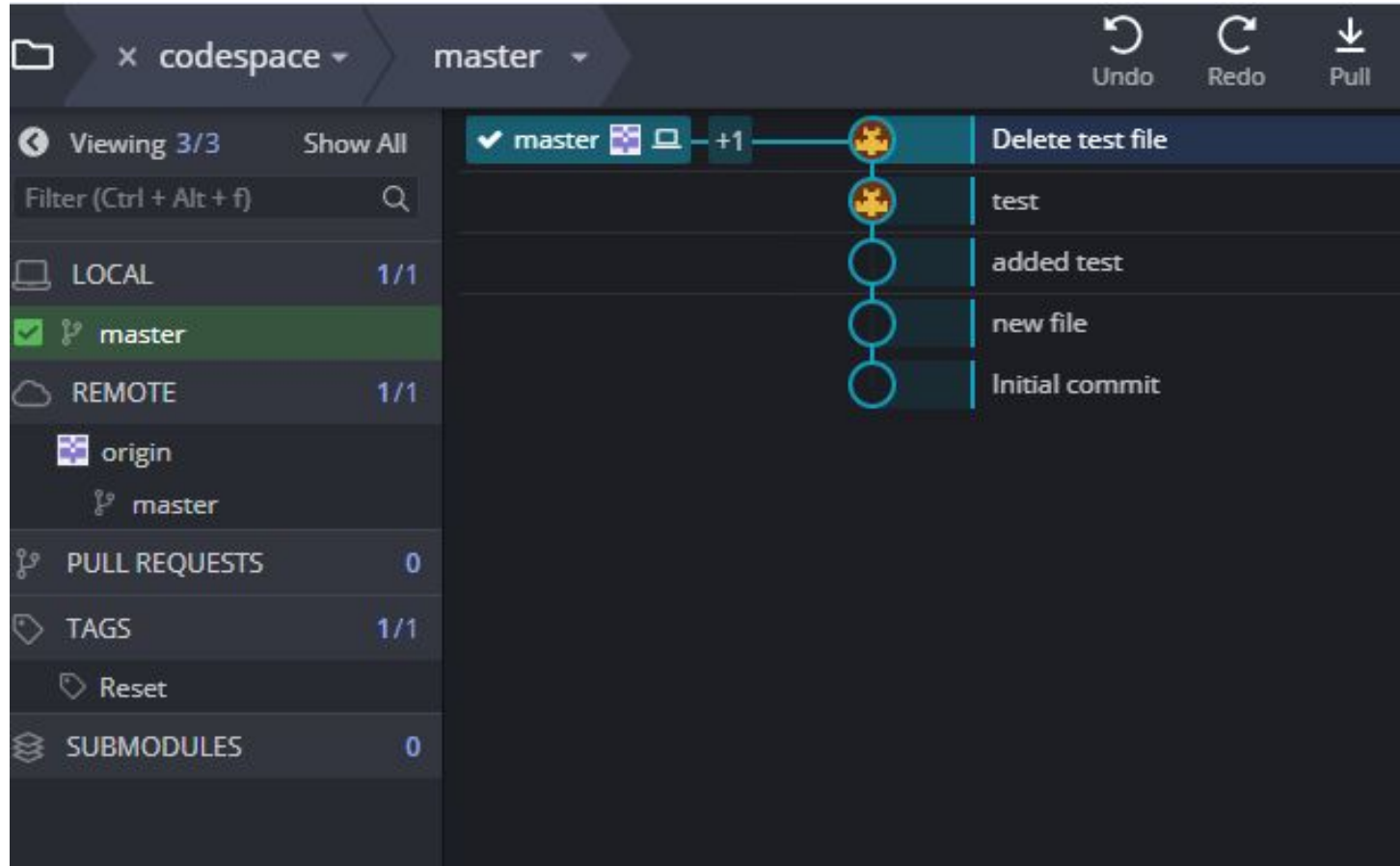
commit 085bb3bcb608e1e8451d4b2432f8ecbe6306e7e7
Author: Scott Chacon <schacon@gee-mail.com>
Date:   Sat Mar 15 16:40:33 2008 -0700

    removed unnecessary test

commit a11bef06a3f659402fe7563abf99ad00de2209e6
Author: Scott Chacon <schacon@gee-mail.com>
Date:   Sat Mar 15 10:31:28 2008 -0700

    first commit
```

git tag



```
$ git tag  
v0.1  
v1.3
```