

Taller de Git, GitHub y GitFlow

Ejercicio 1

08 de marzo de 2017

Autores: Alfredo González Gaviña,

Walter Alejandro Moreno Ramírez y María Susana Ávila García.

El objetivo de este ejercicio que el estudiante comprenda el flujo de trabajo necesario para trabajar con git. Se darán ejemplos prácticos y al final de este documento se plantea un ejercicio que se sugiere, el estudiante intente resolver solo, utilizando los conocimientos que los ejercicios de practica le han provisto.

Al final de esta práctica el estudiante será capaz de comprender el proceso de ramificación según el modelo “*feature Branch*” así como el proceso de fusiones por diversas estrategias, la creación y gestión de repositorios remotos y finalmente el envío y actualización de repositorios locales a partir de repositorios remotos.

1.- Crear un repositorio remoto y conectarlo.

En esta parte se muestran el proceso de creación y configuración de un repositorio remoto

- Paso 1. Crear un nuevo repositorio remoto.
Diríjase hasta la dirección web <https://github.com/> y acceda con su usuario de github, o bien con su correo electrónico (véase figura 1)

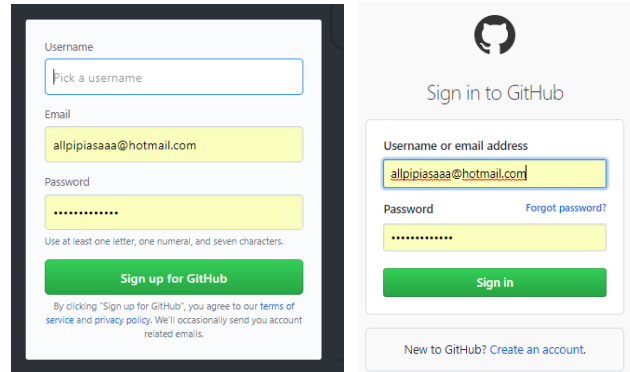


Fig. 1 inicio de sesión en Github.

- Paso 2. Creación de nuevo repositorio remoto.

Cree un nuevo repositorio remoto desde la interfaz web que provee github (véase figura 2). Llame a su nuevo repositorio “Taller3G” y seleccione la opción “Public”. Si desea puede agregar una descripción en el campo “Description”.

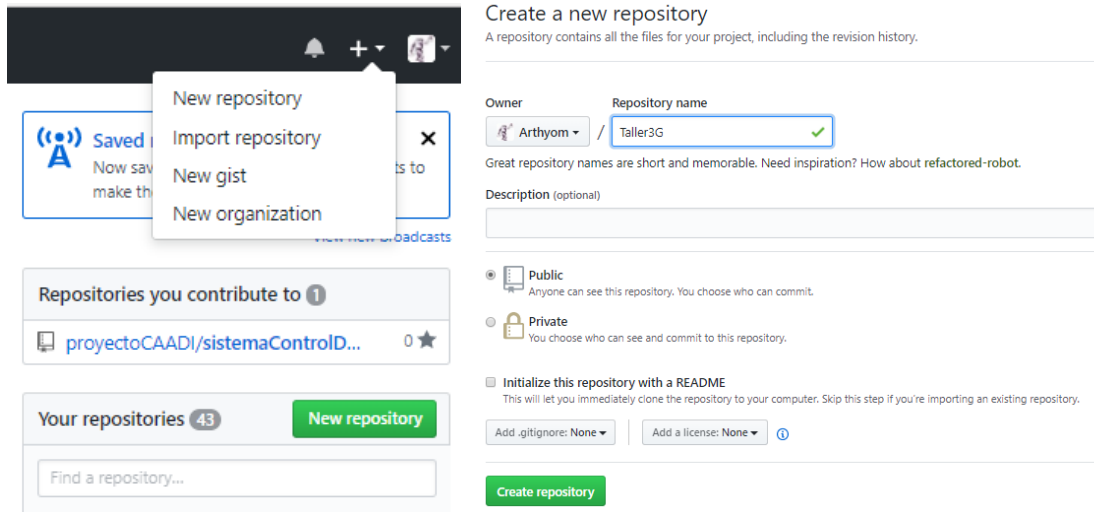


Fig. 2 Creación de un nuevo repositorio desde la interfaz de github.

- Paso 3. Conectar con un remoto.

Copie el enlace que aparece en la parte superior de la pantalla y siga las instrucciones que aparecen en la parte inferior (Véase figura 3).



Fig 3 Pasos iniciales para trabajar con un repositorio remoto.

- Paso 4. Conexión de un repositorio local con un repositorio remoto.
Ahora que ya tiene un reposito remoto para alojar su proyecto debe conectarlo con su repositorio local, para eso usara los comandos:

```
$ git init <nombreRepo>  
$ git remote add <nombreRemoto> <URL>
```

Ejemplo: cree un repositorio remoto desde su terminal usando el comando git init "taller3G" si ya dispone de un repositorio local puede usar el mismo repositorio.

```
$ git init taller3G
```

Use la terminal para moverse hasta el repositorio local que va a utilizar, en caso de no estar ya dentro del directorio "taller3G":

```
$ cd taller3G  
C:\Users\Frodo\Desktop\taller3G>
```

Fig. 4. Salida esperada en la termina una vez dentro del directorio "taller3G".

Conecte su repositorio local con el repositorio remoto mediante el comando:

```
$ git remote add <nombreRemoto> <URL>
```

Ejemplo: Escriba en la terminal la siguiente línea para conectar su repositorio local con el repositorio remoto que creo en el paso 1, note como esta información es desplegada una vez que creo su repositorio remoto en la interfaz de github (Véase figura 5) no recibirá ninguna salida, pero puede usar el comando "git remote" para listar todos los repositorios remotos que tiene en su repositorio local.

```
$ git remote add origin https://github.com/Arthyom/Taller3G.git
```

```
C:\Users\Frodo\Desktop\taller3G>git remote add origin https://github.com/Arthyom/Taller3G.git  
C:\Users\Frodo\Desktop\taller3G>
```

```
C:\Users\Frodo\Desktop\taller3G>git remote -v  
origin https://github.com/Arthyom/Taller3G.git (fetch)  
origin https://github.com/Arthyom/Taller3G.git (push)
```

Fig 5. Conectando un remoto y listando todos los remotos disponibles.

2.- Commits, Ramas, fusiones y remotos.

En esta parte se muestran el proceso de creación de commits y de la creación de ramas siguiendo el modelo feature Branch.

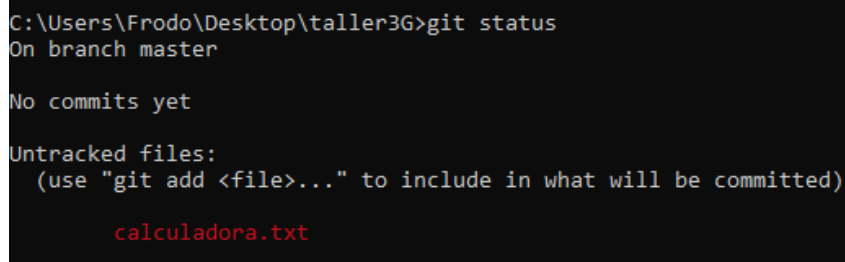
- Paso 1. Agregar archivos.

Crearemos un archivo de texto llamado “calculadora.txt” y lo agregaremos al índice en la rama master para posteriormente confirmar los cambios mediante un commit. Los comandos que usaremos serán:

\$ git add <nombreArchivo>

\$ git commit -m “mensaje”

Ejemplo: cree un archivo llamado “calculadora.txt” usando su editor de texto favorito y guárdelo en la carpeta de su repositorio local, utilice el comando “git status” para visualizar el archivo nuevo desde su terminal (véase figura 7A). Utilice los comandos “git add .” y *git commit -m “commit inicial en master”* para preparar los cambios que serán confirmados (Véase figura 7B).



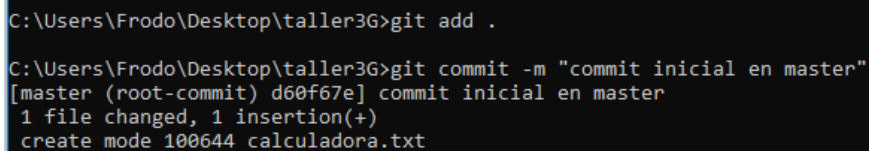
```
C:\Users\Frodo\Desktop\taller3G>git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        calculadora.txt
```

A)



```
C:\Users\Frodo\Desktop\taller3G>git add .
C:\Users\Frodo\Desktop\taller3G>git commit -m "commit inicial en master"
[master (root-commit) d60f67e] commit inicial en master
1 file changed, 1 insertion(+)
create mode 100644 calculadora.txt
```

B)

Fig 7. Creación de ramas y listado de las mismas.

- Paso 2. Primer Push.

Enviaremos los cambios ya confirmados de la rama master hasta el repositorio remoto, utilizaremos el comando.

\$ git push -u [nombreRemoto] [nombreRama]

Ejemplo: Escriba en su terminal el siguiente comando:

\$ git push -u origin master

La terminal le solicitará confirmar su correo o nombre de usuario de github así como

su contraseña y posteriormente enviara los cambios hasta el repositorio remoto (Vease figura 8)

```
C:\Users\Frodo\Desktop\taller3G>git push -u origin master
fatal: HttpRequestException encountered.
Error al enviar la solicitud.
Username for 'https://github.com': arthyom
Password for 'https://arthiom@github.com':
Counting objects: 3, done.
Writing objects: 100% (3/3), 251 bytes | 62.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/Arthyom/Taller3G.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

Fig 8. Salida esperada para un push.

Git le mostrara como salida una gran cantidad de información donde entre otras cosas de indica: 1) Ramas creadas, 2) Nombre del commit, 3) Estadísticas de información. Puede verificar los cambios que ahora se encuentran en el repositorio remoto accediendo a la URL donde fue creado (véase figura 9).

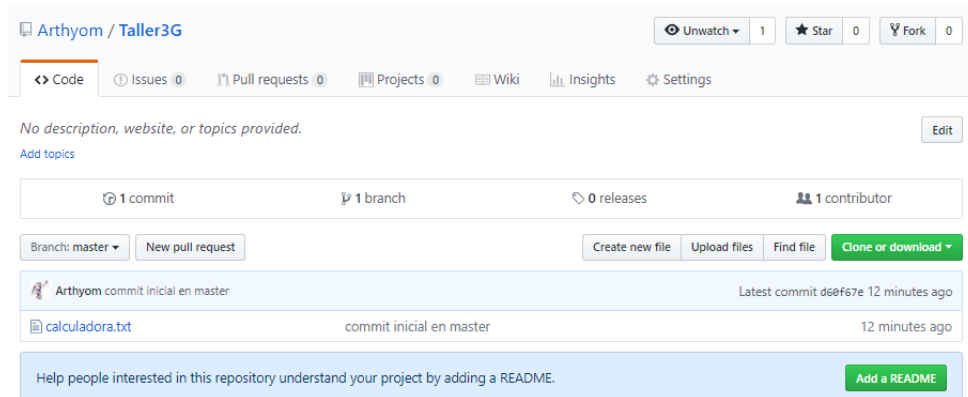


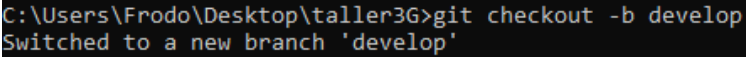
Fig 9. Repositorio remoto actualizado con el primer commit.

- **Paso 3. El modelo feature branch.**

Siguiendo el modelo feature Branch crearemos una nueva rama “develop” para guardar las versiones en desarrollo de nuestro repositorio local, también crearemos funcionalidades nuevas usando ramas alternas. Cabe señalar que la rama develop se deriva de master mientras que las ramas de funcionalidad siempre deben derivarse a partir de develop. Usaremos los comandos:

```
$git checkout <nombrerama>
$ git branch <nombrerama>
$ git checkout -b <nombreRama>
$ git merge <nombreRama>
```

Ejemplo: cree una nueva rama y cámbiese a esta usando el comando “git checkout -b develop”, el comando git checkout -b es equivalente a los comandos “git Branch <nombreRama>” y “git checkout <nombreRama>.” (véase figura 10)



```
C:\Users\Frodo\Desktop\taller3G>git checkout -b develop
Switched to a new branch 'develop'
```

Fig 10. Creación de la rama develop

A continuación, crearemos una rama de funcionalidad para cada nueva operación que realiza una calculadora básica, es decir crearemos una rama para la suma, otra para la resta y así sucesivamente. Nombre a cada rama de funcionalidad con la siguiente nomenclatura “feature-operación” por ejemplo para la suma el nombre adecuado podría ser “feature-suma”. En Cada nueva operación deberá indicar brevemente la operación que está desarrollando después guardar los cambios en su archivo, agregarlos al índice y confirmarlos para cada rama de funcionalidad, también deberá enviar los cambios de cada rama hasta el repositorio remoto. Una vez que termine de definir una operación debe integrarla con la rama “develop” mediante una fusión con el comando “git merge” y finalmente debe empujar la rama develop hasta el repositorio remoto con el comando “git push”. Asegúrese de fusionar develop con master una vez que termine el ejercicio así como también debe enviarlo al repositorio remoto para generar una versión estable y lista para salir al mercado (véase figura 11).

```
calculadora.txt x
1  Hola esta es una calculadora
2
3  Suma
4  la suma o adición es una operación lineal donde el valor de dos números
5  es aumentado.
```

A)

```
C:\Users\Frodo\Desktop\taller3G>git checkout -b feature-suma
Switched to a new branch 'feature-suma'

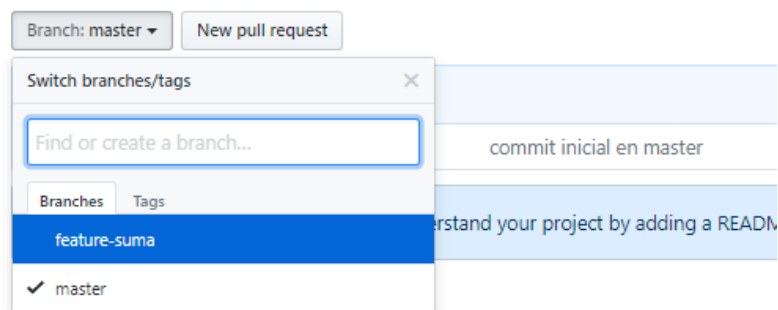
C:\Users\Frodo\Desktop\taller3G>git status -s
 M calculadora.txt

C:\Users\Frodo\Desktop\taller3G>git add .

C:\Users\Frodo\Desktop\taller3G>git commit -m "definiendo operación suma"
[feature-suma bcd2270] definiendo operación suma
 1 file changed, 5 insertions(+), 1 deletion(-)
```

B)

```
C:\Users\Frodo\Desktop\taller3G>git push -u origin feature-suma
fatal: HttpRequestException encountered.
Error al enviar la solicitud.
Username for 'https://github.com': arthyom
Password for 'https://arthiom@github.com':
Counting objects: 3, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 344 bytes | 57.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/Arthyom/Taller3G.git
 * [new branch]      feature-suma -> feature-suma
Branch 'feature-suma' set up to track remote branch 'feature-suma' from 'origin'.
```



C)

```
C:\Users\Frodo\Desktop\taller3G>git checkout develop
Switched to branch 'develop'

C:\Users\Frodo\Desktop\taller3G>git merge feature-suma
Updating d60f67e..324f6ab
Fast-forward
 calculadora.txt | 6 +++++-
 1 file changed, 5 insertions(+), 1 deletion(-)
```

D)

Fig 11.

- A) Creación de definiciones para cada operación en el archivo calculadora.txt
- B) Creación de la rama "feature-suma" a partir de la rama develop, se agregan los cambios al índice con "git add" y después se confirman con "git commit".
- C) Envío de cambios en la rama "feature-suma" del repositorio local hasta el repositorio remoto, puede verificarlo en su navegador web.
- D) Fusión de la rama "feature-suma" con la rama develop

Repita este proceso para cada una de las operaciones restantes.

