

La guía básica de Git y Github para principiantes



Git es el mejor sistema de control de versiones del mundo y con esta guía básica aprenderás comandos simples para poder empezar a trabajar con tu equipo y subir tus proyectos de forma fácil y rápida.

En este post hablaremos de que es Git y Github, además de como usar la terminal de Git Bash para los usuarios de Windows y así poder trabajar colaborativamente con tu equipo.

¿Qué es Git ?

Es un sistema de control de versiones, es distribuido, es decir que múltiples personas pueden trabajar en equipo, es open source y también se adapta a todo tipo de proyectos desde pequeños hasta grandes, además, se pueden fusionar archivos, guarda una línea de tiempo a lo largo de todo el proyecto.

¿Qué es Github?

Es una plataforma de desarrollo colaborativo, o también llamada la red social de los desarrolladores donde se alojan los repositorios, el código se almacena de forma pública pero se puede hacer privado con una cuenta de pago.

Aprendiendo sobre comandos

Empezaremos con los 6 comandos mas importantes en consola para todo principiante.

1. **pwd** : Para ubicarme en que directorio me encuentro.

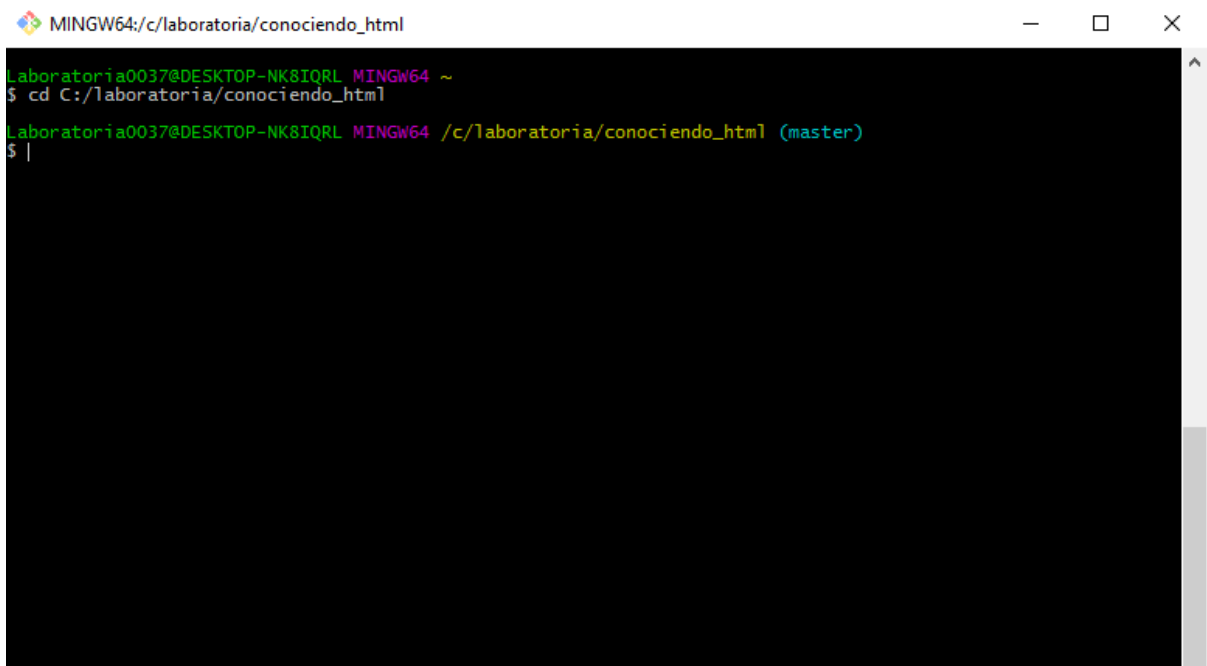


```
MINGW64:/c/Users/Laboratoria0037
Laboratoria0037@DESKTOP-NK8IQL MINGW64 ~
$ pwd
/c/Users/Laboratoria0037
Laboratoria0037@DESKTOP-NK8IQL MINGW64 ~
$ |
```

Nota:

..*(Punto punto)* : Subir un nivel al directorio moverse a un lugar fuera.

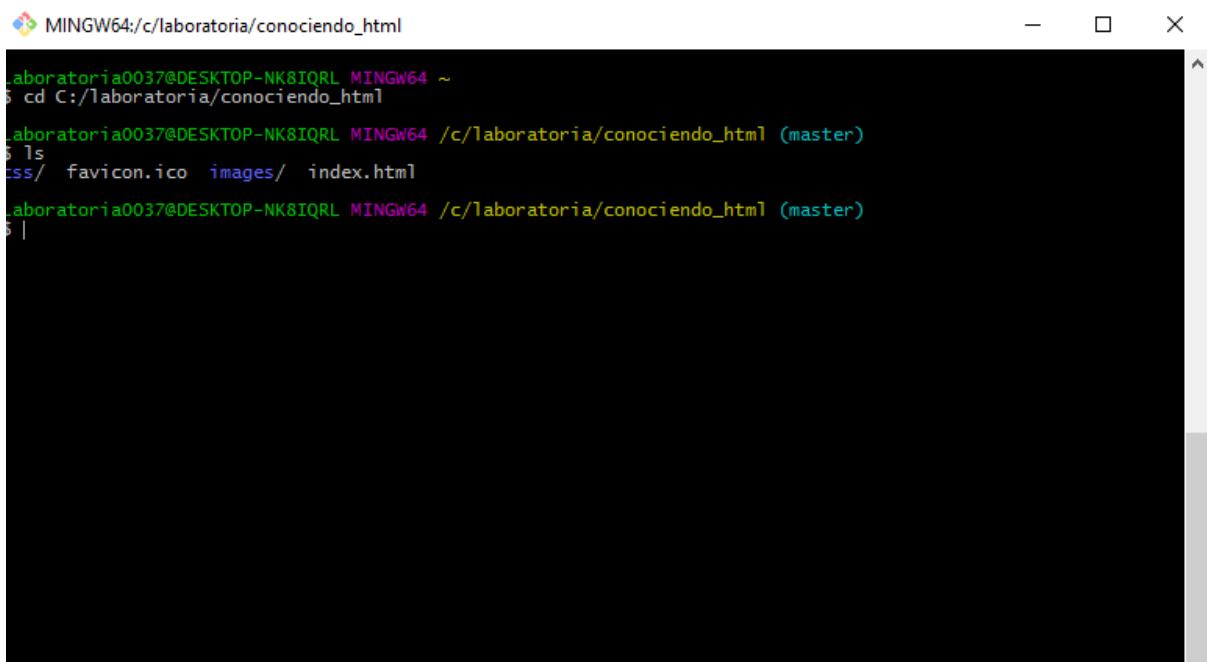
2. cd : Para situarse dentro del directorio.



A terminal window titled 'MINGW64:/c/laboratoria/conociendo_html'. The prompt is 'Laboratoria0037@DESKTOP-NK8IQR L MINGW64 ~'. The user enters 'cd C:/laboratoria/conociendo_html'. The prompt changes to 'Laboratoria0037@DESKTOP-NK8IQR L MINGW64 /c/laboratoria/conociendo_html (master)'. The user enters a dollar sign '\$' followed by a vertical bar '|', indicating the command is being typed.

```
MINGW64:/c/laboratoria/conociendo_html
Laboratoria0037@DESKTOP-NK8IQR L MINGW64 ~
$ cd C:/laboratoria/conociendo_html
Laboratoria0037@DESKTOP-NK8IQR L MINGW64 /c/laboratoria/conociendo_html (master)
$ |
```

3. ls : Para listar todo lo que está dentro de donde nos situamos



A terminal window titled 'MINGW64:/c/laboratoria/conociendo_html'. The prompt is 'Laboratoria0037@DESKTOP-NK8IQR L MINGW64 ~'. The user enters 'cd C:/laboratoria/conociendo_html'. The prompt changes to 'Laboratoria0037@DESKTOP-NK8IQR L MINGW64 /c/laboratoria/conociendo_html (master)'. The user enters 'ls'. The output is 'css/ favicon.ico images/ index.html'. The user enters a dollar sign '\$' followed by a vertical bar '|', indicating the command is being typed.

```
MINGW64:/c/laboratoria/conociendo_html
Laboratoria0037@DESKTOP-NK8IQR L MINGW64 ~
$ cd C:/laboratoria/conociendo_html
Laboratoria0037@DESKTOP-NK8IQR L MINGW64 /c/laboratoria/conociendo_html (master)
$ ls
css/ favicon.ico images/ index.html
Laboratoria0037@DESKTOP-NK8IQR L MINGW64 /c/laboratoria/conociendo_html (master)
$ |
```

4. ll : Para listar de una forma más grande

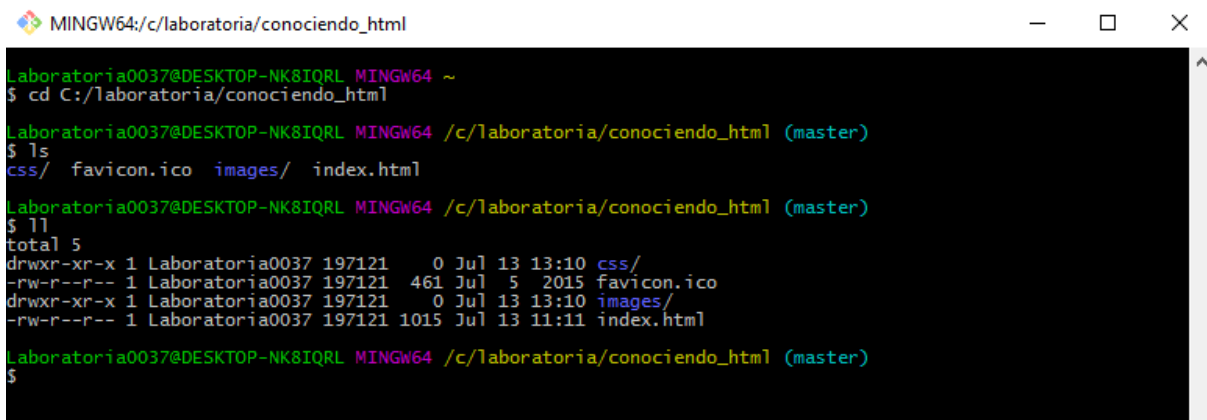
Qué entendemos por -rw-r-r — 1

READ (r)

WRITE (w)

EXECUTE(x)

Son permisos para los directorios y archivos.

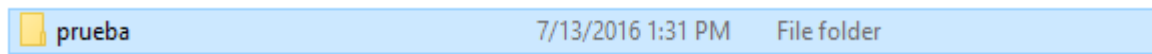


```
MINGW64:/c/laboratoria/conociendo_html
Laboratoria0037@DESKTOP-NK8IQL MINGW64 ~
$ cd C:/laboratoria/conociendo_html
Laboratoria0037@DESKTOP-NK8IQL MINGW64 /c/laboratoria/conociendo_html (master)
$ ls
css/  favicon.ico  images/  index.html
Laboratoria0037@DESKTOP-NK8IQL MINGW64 /c/laboratoria/conociendo_html (master)
$ ll
total 5
drwxr-xr-x 1 Laboratoria0037 197121  0 Jul 13 13:10 css/
-rw-r--r-- 1 Laboratoria0037 197121 461 Jul  5 2015 favicon.ico
drwxr-xr-x 1 Laboratoria0037 197121  0 Jul 13 13:10 images/
-rw-r--r-- 1 Laboratoria0037 197121 1015 Jul 13 11:11 index.html
Laboratoria0037@DESKTOP-NK8IQL MINGW64 /c/laboratoria/conociendo_html (master)
$
```

5. mkdir : Para crear directorios.

```
Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/conociendo_html (master)
$ mkdir prueba
```

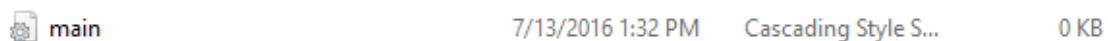
Resultado:



6 . touch : Para crear los ficheros archivos.

```
Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/conociendo_html (master)
$ touch prueba/main.css
```

Resultado:



Primer uso de Git Bash

1.git init : Todo lo que se escriba dentro de la carpeta va a estar dentro del **.git**

Allí va el historial de versiones y está oculto dentro de la carpeta.

→ Solo una vez.

2. git add : Lleva el control de los archivos que se agregan luego de escribir ese comando.

2.1 git add . : Es para todo agregar pero no es recomendable.

3. git commit : Comando que indica que esta lista alguna funcionalidad para que sea una versión del código.

4. git commit -m "Mi primer commit"

→ Se repite n veces cada vez se cambia el código.

Nota :

- *Se tiene que detallar muy bien lo que se pone en el comentario es una buena práctica para que nuestro*

equipo colaborativo nos pueda entender y solucionar errores en caso sucediera.

- *Cuando coinciden las líneas entre una persona y la otra pueden ocurrir conflictos.*

Push

Básicamente lo que realiza un push es publicar lo que se encuentra en nuestro servidor local y llevarlo al servidor remoto de Github

El push empuja lo de nuestro servidor local al servidor remoto en github

Pull

El pull trae los cambios de nuestro repositorio remoto y los actualiza al repositorio local.

5. git remote add origin [URL DEL REPOSITORIO EN GITHUB]

6. git push origin master : Es el nombre que se le pone al repositorio remoto al que se conecta.

¿Qué es un Branch ?

Es una rama diferente a la rama master que tenemos y se utiliza para trabajar de forma colaborativa y eficaz con esto podremos asignarles tareas y ordenar mejor nuestro proyecto.

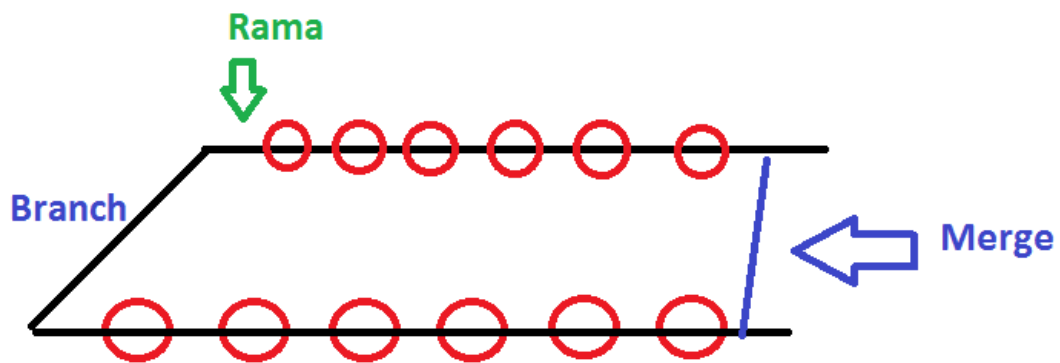
¿Qué es un Merge?

Es una fusión entre ramas.

Los conceptos entre Branch y Merge suelen ser difíciles de aprender pero con los ejemplos podrás tener una mejor idea de que es y para qué utiliza.

Utilizando el comando Branch

git push origin [Nombre de la Rama]



Ejemplo de Merge

1. Situar en la rama de producción en este caso master:
git checkout master.
3. Realizar un merge con la rama que se desea incluir : **git merge portada**
5. Publicarlo : **git push origin master**

Como Publicar mi proyecto en Github

La práctica es mucho mejor que la teoría estos son los primeros pasos para empezar a familiarizarse con la consola de Git y trabajar en proyectos que se alojen en Github.

Pasos:

1. Situarse en la carpeta donde esta nuestro contenido con cd

```
Laboratoria0037@DESKTOP-NK8IQL MINGW64 ~  
$ cd C:/repository/ConociendoHTML  
Laboratoria0037@DESKTOP-NK8IQL MINGW64 /c/repository/ConociendoHTML (master)  
$
```

2. Realizar un **git init**
3. Realizar un add (**git add .**)

Realizando mi Primer commit

Al escribir el commit cuando lo utilizamos por primera vez nos pide configurar las credenciales

4. Configurando mis credenciales

```
Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/conociendo_html (master)
$ git commit -m "Primeros pasos en HTML"

*** Please tell me who you are.

Run

    git config --global user.email "you@example.com"
    git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: empty ident name (for <(null)>) not allowed

Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/conociendo_html (master)
$ git config --global user.email "sthefanyflorianog@gmail.com"

Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/conociendo_html (master)
$ git config --global user.name "Sthefany Floriano"
```

2. git commit — m “Primeros pasos en HTML”

```
Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/conociendo_html (master)
$ git commit -m "Primeros pasos en HTML"
[master (root-commit) bb463f7] Primeros pasos en HTML
warning: LF will be replaced by CRLF in index.html.
The file will have its original line endings in your working directory.
5 files changed, 55 insertions(+)
 create mode 100644 css/style.css
 create mode 100644 favicon.ico
 create mode 100644 images/cara.jpg
 create mode 100644 index.html
 create mode 100644 prueba/main.css

Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/conociendo_html (master)
$ |
```

3. Utilizar un remote add origin para conectar nuestro repositorio local con el github la url tiene que ser exactamente igual al url que creamos en el github.

```
Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/conociendo_html (master)
$ git remote add origin https://github.com/sthecrash/proyecto-laboratoria.git
```

4. Realizamos un push

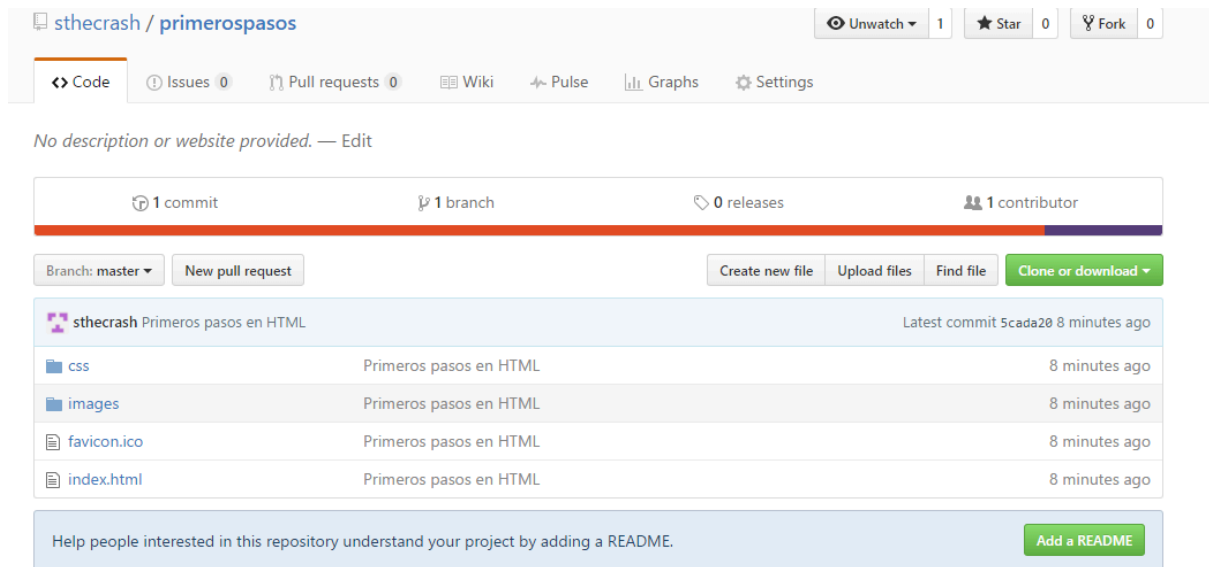
```
Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/conociendo_html (master)  
$ git push origin master
```

Pide credenciales :

Poner nombre de usuario de github y clave de github

5. Volvemos a hacer el commit

Listo el Repositorio ha sido publicado en Github.



Publicando mi proyecto en Github Pages

1. Situarse en la carpeta donde está el proyecto con `cd [Ruta]`
2. Para que pueda publicarse tiene que estar dentro de una rama se procede a crearla.

```
Laboratoria0037@DESKTOP-NK8IQR L MINGW64 /c/laboratoria/ConociendoHTML (master)
$ git checkout -b gh-pages
Switched to a new branch 'gh-pages'
```

1. Verificar con **git status**

```
Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/ConociendoHTML (gh-pages)
$ git status
On branch gh-pages
nothing to commit, working directory clean
```

Nota : En mi caso no hay cambios pero en caso los hubiera se usaría un git add

1. Realizar un **git commit -m 'initial commit'**

```
Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/ConociendoHTML (gh-pages)
$ git commit -m 'initial commit'
On branch gh-pages
nothing to commit, working directory clean
```

1. Publicarlo **git push origin gh-pages**

```
Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/ConociendoHTML (gh-pages)
$ git push origin gh-pages
Total 0 (delta 0), reused 0 (delta 0)
To https://github.com/sthecrash/primerospasos.git
 * [new branch]      gh-pages -> gh-pages
```

Información adicional

Si te equivocas al poner el mensaje en el commit se corrige con el — amend

```
Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/ConociendoHTML (master)
$ git commit -m "Icono en carpeta de images" --amend
[master b6f5ca9] Icono en carpeta de images
Date: Thu Jul 14 10:05:50 2016 -0700
2 files changed, 1 insertion(+), 1 deletion(-)
rename favicon.ico => images/favicon.ico (100%)
```

Luego se realiza un pull

```
Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/ConociendoHTML (master)
$ git pull origin master
From https://github.com/sthecrash/primerospasos
 * branch          master      -> FETCH_HEAD
Merge made by the 'recursive' strategy.
```

Finalmente un push

```
Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/ConociendoHTML (master)
$ git push origin master
Counting objects: 2, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 367 bytes | 0 bytes/s, done.
Total 2 (delta 1), reused 0 (delta 0)
To https://github.com/sthecrash/primerospasos.git
 f0de4d9..df8d03f  master -> master
```

Y listo corregi el contenido de mi mensaje

sthecrash Merge branch 'master' of https://github.com/sthecrash/primerospasos		Latest commit df8d03f 6 minutes ago
css	Primeros pasos en HTML	20 hours ago
images	Icono en carpeta de images	6 minutes ago
index.html	Icono en carpeta de images	6 minutes ago

Ejemplo : Subiendo Proyecto a Github

Luego de realizar los primeros pasos y tener las credenciales correctas ya no es necesario realizar los pasos anteriores con este ejemplo aprenderás como Git puede ser más sencillo de lo que imaginas.



MINGW64:/c/laboratoria/viendo-cajas

Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/viendo-cajas



Situandose en la carpeta

Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/viendo-cajas (master)
\$ git add .



Agregando los ficheros

```
Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/viendo-cajas (master)
$ git status
On branch master

Initial commit


Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

        new file:   images/favicon.ico
        new file:   images/logo.png
        new file:   index.html
```



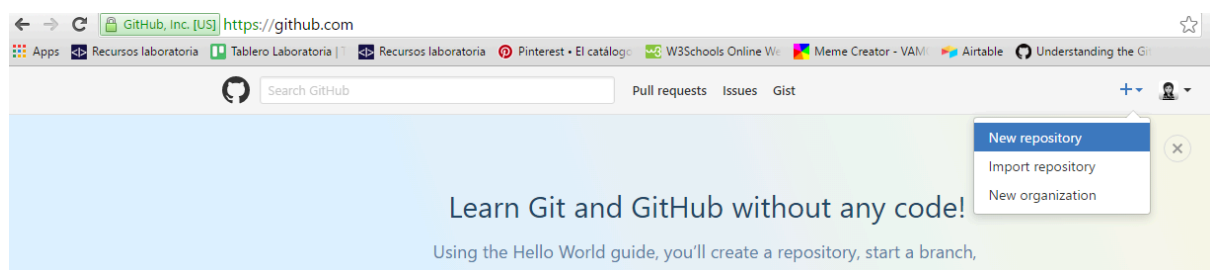
Viendo el estado de mi rep

```
Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/viendo-cajas (master)
$ git commit -m "Primer Commit"
[master (root-commit) a44b77f] Primer Commit
3 files changed, 23 insertions(+)
create mode 100644 images/favicon.ico
create mode 100644 images/logo.png
create mode 100644 index.html
```




Realizando un commit

Luego de esto se va a la página de github (<https://github.com>) y se crea un nuevo repositorio.



Utilizamos el comando git remote + la url del repositorio que creamos en la página de github.

```
Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/viendo-cajas (master)  
$ git remote add origin https://github.com/sthecrash/viendo-cajas.git
```



Solo se realiza una vez

Ahora realizamos un push para que aparezcan los cambios en el github

```
Laboratoria0037@DESKTOP-NK8IQRL MINGW64 /c/laboratoria/viendo-cajas (master)  
$ git push -u origin master  
Counting objects: 6, done.  
Delta compression using up to 4 threads.  
Compressing objects: 100% (6/6), done.  
Writing objects: 100% (6/6), 13.05 KiB | 0 bytes/s, done.  
Total 6 (delta 0), reused 0 (delta 0)  
To https://github.com/sthecrash/viendo-cajas.git  
* [new branch]      master -> master  
Branch master set up to track remote branch master from origin.
```

Clonando y Colaborando un repositorio de Github


Si bien vimos como clonar esta vez será diferente por que trabajaremos con ramas(branch) para poder colaborar de forma eficiente en nuestro proyecto.

1. Situarse en el directorio donde traeremos nuestro repositorio


en este caso lo pondré en C:/

git clone [URL DEL PROYECTO]

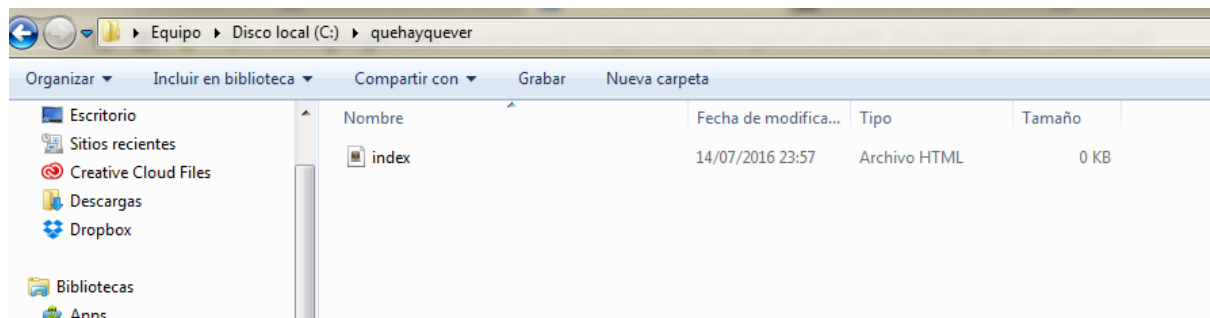
Nota: Se pone la Url de github del proyecto que queremos colaborar.

 GitHub, Inc. [US] <https://github.com/teresalara/quehayquever>

2.

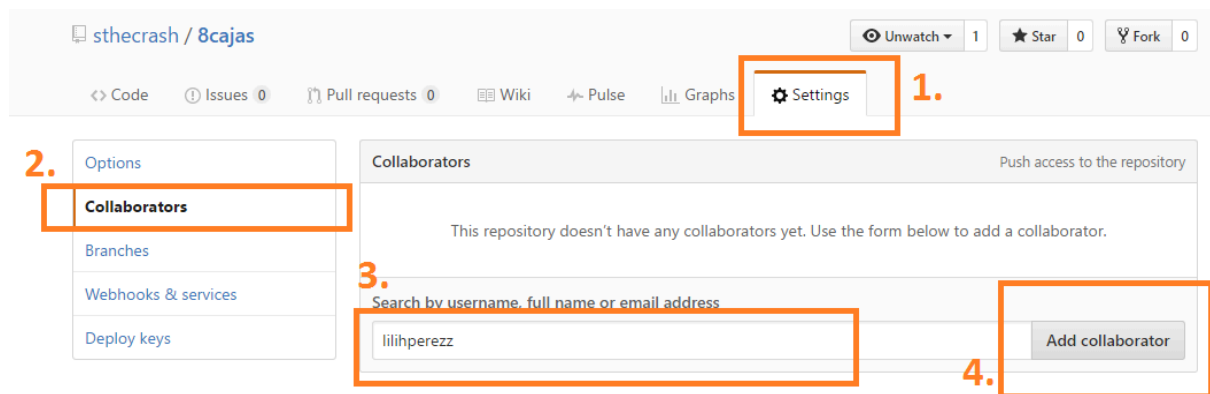
 MINGW64:/c
Laboratoria0037@DESKTOP-NK8IQL MINGW64 /c
\$ git clone https://github.com/teresalara/quehayquever
Cloning into 'quehayquever'...
remote: Counting objects: 136, done.
remote: Compressing objects: 100% (120/120), done.
Receiving objects: 63% (87/136), 13.21 MiB | 40.00 KiB/s

Resultado :



Colaborando en Github

Es importante que para colaborar puedas tener permisos y esto se realiza de la siguiente forma:



3. Nos situamos en la carpeta y ya estamos conectados con la master

```
STHEFANY@STHEFANY-PC MINGW64 /c
$ cd c:/quehayquever/

STHEFANY@STHEFANY-PC MINGW64 /c/quehayquever (master)
$ ls -all
total 44
drwxr-xr-x 1 STHEFANY 197121 0 jul 14 23:57 ./
drwxr-xr-x 1 STHEFANY 197121 0 jul 14 23:57 ../
drwxr-xr-x 1 STHEFANY 197121 0 jul 14 23:57 .git/
-rw-r--r-- 1 STHEFANY 197121 0 jul 14 23:57 index.html
```

4. Creare una rama(branch) para colaborar desde mi usuario sthecrash

```
STHEFANY@STHEFANY-PC MINGW64 /c/quehayquever (master)
$ git checkout -b sthefanyfloriano
Switched to a new branch 'sthefanyfloriano'

STHEFANY@STHEFANY-PC MINGW64 /c/quehayquever (sthefanyfloriano)
$
```

5. Haré cambios a mi index.html para realizar un commit

```
index.html
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <meta charset="utf-8">
5          <title>Probando un nuevo branch</title>
6      </head>
7      <body>
8          <p> Estoy probando mi rama :D <p>
9      </body>
10 </html>
```

6. Veo los cambios

```
STHEFANY@STHEFANY-PC MINGW64 /c/quehayquever (sthefanyfloriano)
$ git status
On branch sthefanyfloriano
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   index.html

no changes added to commit (use "git add" and/or "git commit -a")
```

7. add

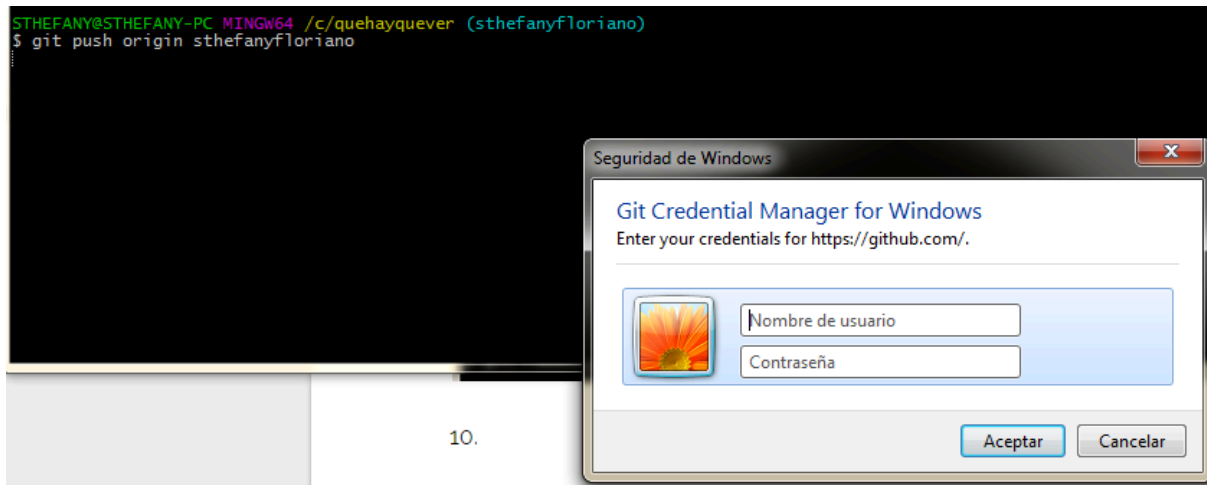
```
STHEFANY@STHEFANY-PC MINGW64 /c/quehayquever (sthefanyfloriano)
$ git add *.html
```

8. commit

```
STHEFANY@STHEFANY-PC MINGW64 /c/quehayquever (sthefanyfloriano)
$ git commit -m "Primer commit de mi rama"
[sthefanyfloriano 3f9397c] Primer commit de mi rama
1 file changed, 10 insertions(+)
```

10. Hacer el push

11. llenar con el nombre y contraseña del github



12. Listo acabo de subir mi rama al github

```
STHEFANY@STHEFANY-PC MINGW64 /c/quehayquever (sthefanyfloriano)
$ git push origin sthefanyfloriano
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 386 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/teresalara/quehayquever
 * [new branch]      sthefanyfloriano -> sthefanyfloriano
```

Resultado :

Default branch

master Updated 5 hours ago by teresalara

Default

Your branches

sthefanyfloriano Updated 14 minutes ago by fany1995fg

0 | 1

New pull request



Active branches

sthefanyfloriano Updated 14 minutes ago by fany1995fg

0 | 1

New pull request

