Github

Crear un nuevo repositorio (que para nuestro caso será el repositorio remoto)

A screenshot of a computer

Description automatically generated

- Una vez allí, se le da un nombre (obligatorio), una descripción (opcional), y lo podemos dejar público, y tampoco agregar el archivo inicial readme.md

Una vez creado de forma exitosa, aparecerá algo como esto

A screenshot of a computer

Description automatically generated

Que se confirma, dado que al inicio aparece el nombre mío junto con el nombre del repositorio (PruebaCursoGit).

En la ventana **Quick setup — if you’ve done this kind of thing before** aparecen una serie de pasos que se hacen generalmente para configurarlo, pero eso ya lo hemos hecho desde la terminal cuando llevamos nuestros archivos de la fase 1 a la 2 y a la 3.

- El único paso que no hemos hecho son las dos últimas líneas de esta ventana, tiene que ver con :

git remote add origin <https://github.com/JhonPenafiel/PruebaCursoGit.git>

%% desde luego copiar el comando anterior en la terminal.

%% del último comando, el nombre que se generó al repositorio es origin, pero se pudo haber escrito: git remote add PruebaCursoGit <https://github.com/JhonPenafiel/PruebaCursoGit.git>, pero siempre se utiliza la primera convención origin (la segunda convención es upstream)

- Para verificar que se ha agregado satisfactoriamente:

git remote -v

- Lo siguiente es hacer el push. Es decir, enviar archivos desde nuestro repositorio local a nuestro repositorio remoto.

git push origin master

% que produce-> origin https://github.com/JhonPenafiel/PruebaCursoGit.git (fetch)

origin https://github.com/JhonPenafiel/PruebaCursoGit.git (push)

%% ya se explicó lo de origin. Master tiene que ver con la rama. Por defecto, hemos trabajado en la rama master

%% se lee encontes: pushea nuestro código a este repositorio origin, y vas a enviar lo que está en la rama master

Ahora sólo es cuestión de refrescar mi página en github y vemos como los archivos que pusheamos aparecen

A screenshot of a computer

Description automatically generated

Hagamos una prueba de actualización de un archivo dentro del proyecto

1. Modifique uno de los archivos, ejemplo en archivo.txt ahora coloqué un Hola mundo
2. En el terminal git add archivo.txt
3. git commit -m"arreglando texto"
4. git push origin master
5. Se refresca el github, y si se abre el archivo aparecerá con su última modificación

Si se da click en commit

A screenshot of a computer

Description automatically generated

Y luego en la último de ellos

A screenshot of a computer

Description automatically generated

Vemos que nos mostrará las diferencias

A screenshot of a computer

Description automatically generated

Y esto es lo más poderoso de git.

Ahora para complementar la introducción del git veamos un comando llamado git pull.

Resulta que a veces vamos a trabajar con otras personas, ellos van a hacer cambios y ellos van a hacer push al repositorio remoto y yo necesito halar esos cambios que ellos han hecho, es decir, desde mi repositorio local yo deseo halar los cambios más actuales que se van a encontrar en el repositorio remoto.

Como por ahora no vamos a tener en este ejemplo alguien más compartiendo el repositorio, vamos a cambiar directamente desde el github:

* Dar click en el archivo que quiere editar, por ejemplo en archivo.txt

A screenshot of a computer

Description automatically generated

Hacer click en el lápiz (edit this file). Modificar y guardar (nombre ese commit cambiando texto otra vez)

De esta forma, si se vuelve al repositorio y de nuevo a commit aparecerá

A screenshot of a computer

Description automatically generated

Pero si en la terminal hacemos git log, vemos que sólo tenemos 4 commits. Y eso es porque el repositorio remoto tiene cambios que yo no tengo en mi repositorio local. De esta forma, si abrimos el archivo desde mi compu, vemos que el archivo no ha sido modificado aún.

- Para tal caso debo halar desde el repositorio remoto al local

git pull origin master

automáticamente me actualiza el repositorio local, lo que se verifica con los cambios hechos en ese último archivo.