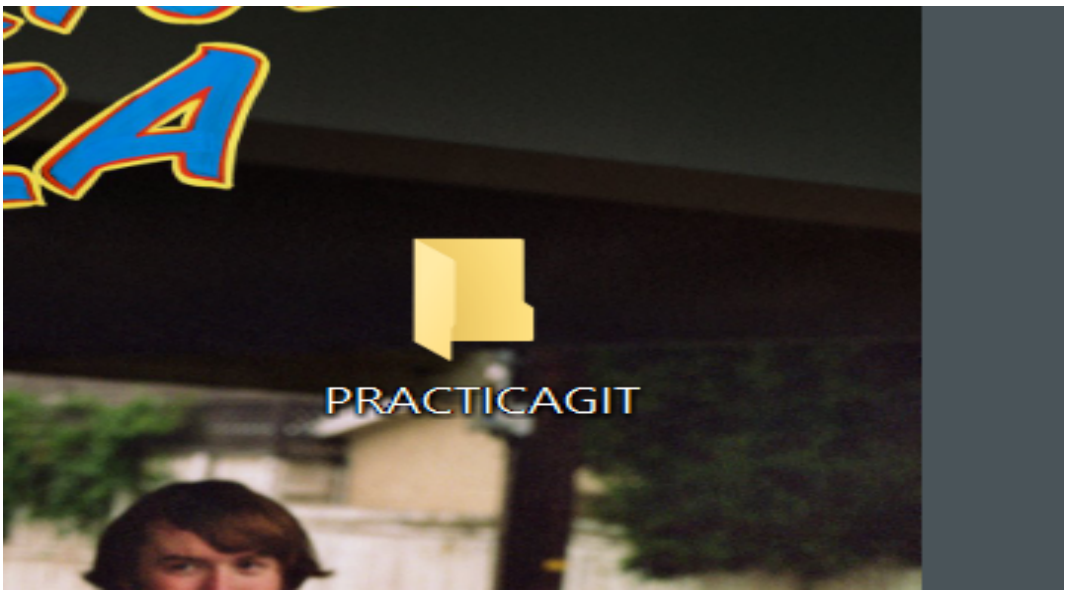
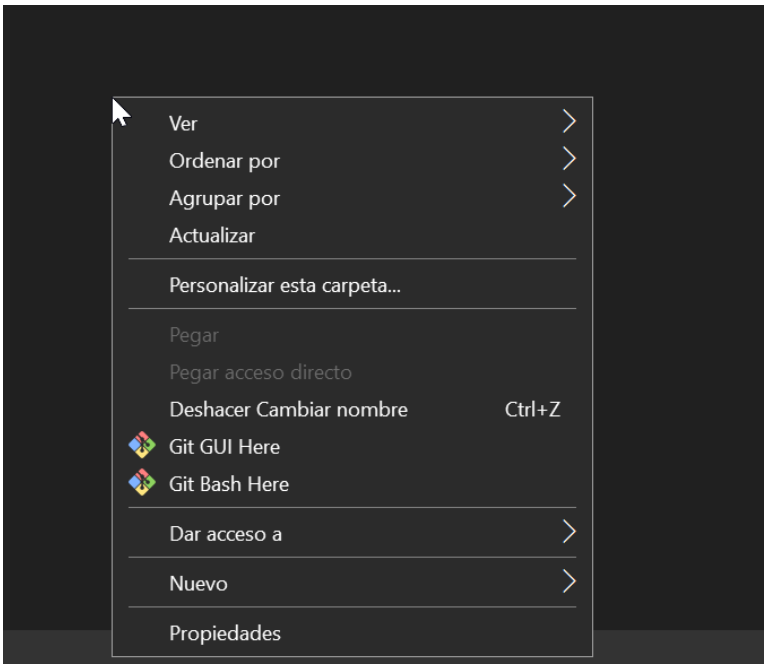
PASO A PASO GIT

En esta guía voy a ir paso a paso desde cero para que puedan hacer la tarea de cargar el cv que nos pidieron en Desarrollo Web y Aplicaciones Digitales en el Instituto ISPC. Aunque es muy básica, nos va a servir para poder hacer la tarea y de paso aprender un poco. Luego ustedes pueden profundizar más en los diferentes comandos que no veremos aquí con videos o con la documentación de git. Vamos a tocar puntos como; Clonar un repositorio de Github, Crear una rama en Git para poder trabajar en ella, subir nuestro cv a nuestra rama y subir esos cambios junto con nuestra rama al repositorio remoto en Github, entre otros. Por lo tanto te servirá de guía por si te olvidas de hacer algo y te ayudará con la tarea.

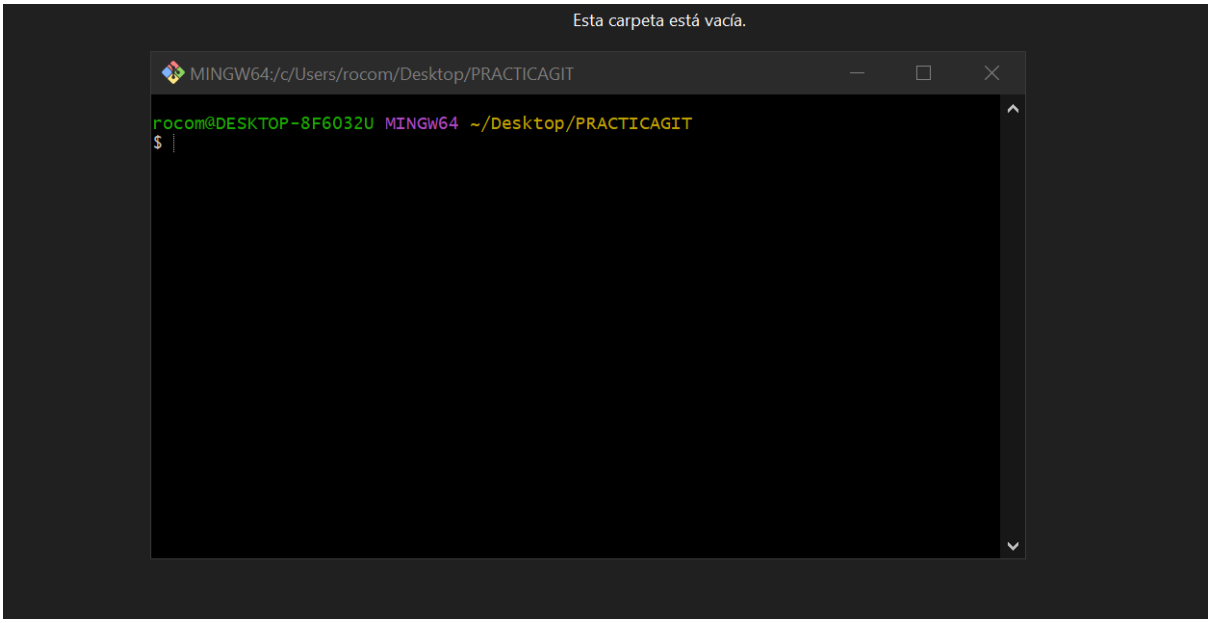
1) Creamos e ingresamos a la carpeta creada en el escritorio que es donde abriremos la consola Git Bash



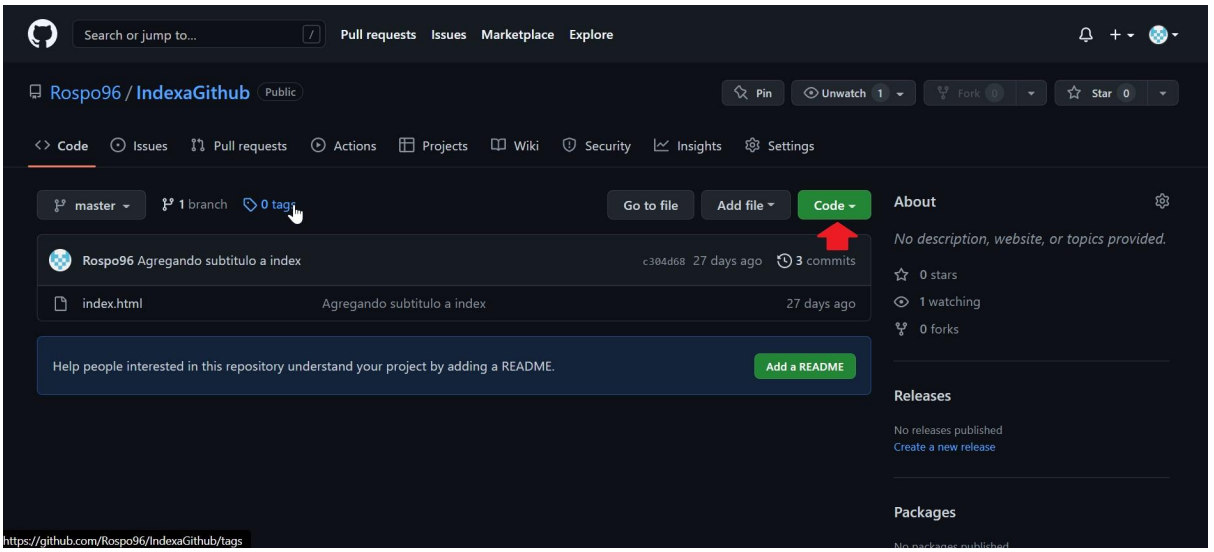
2)Una vez dentro de la carpeta hacemos click derecho y seleccionamos “Git Bash Here”



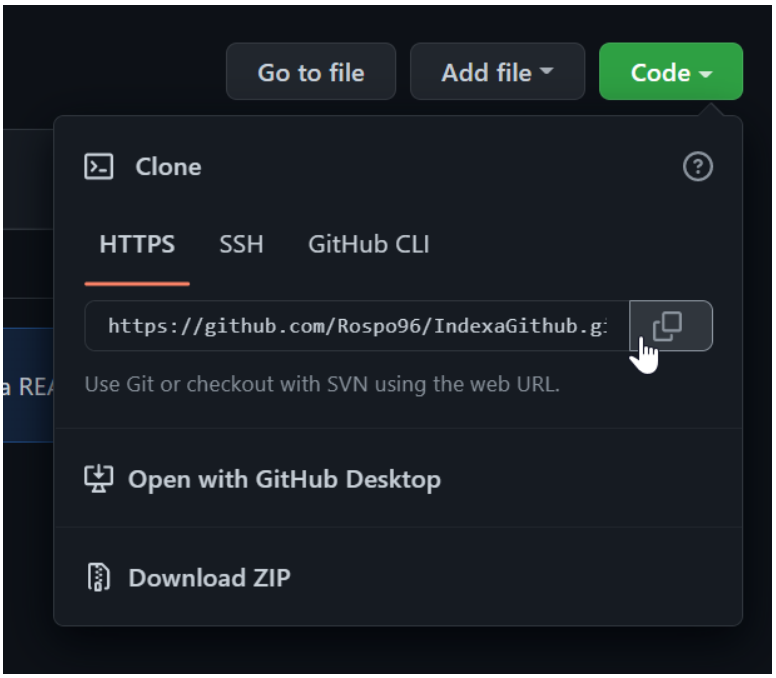
3)Se nos abrirá la consola de Git dentro del directorio o carpeta que hemos creado



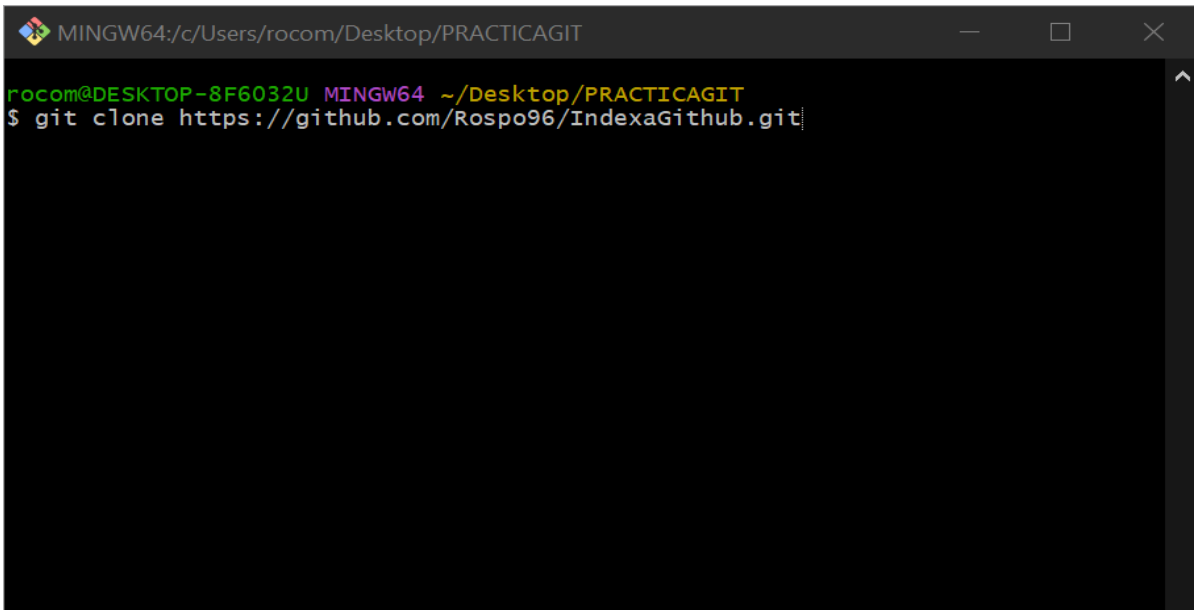
4)Ahora nos vamos al repositorio en GitHub que queramos clonar,sin cerrar la consola de git previamente abierta, y hacemos click en el botón verde “Code”



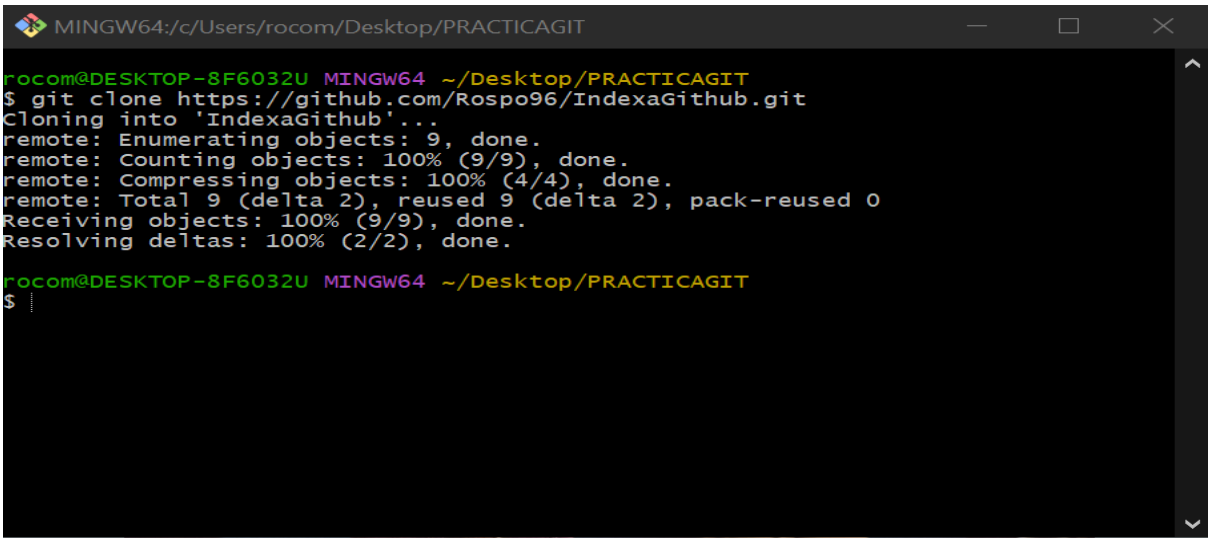
5)Se nos despliega este recuadro y hacemos click donde se ve en la captura. Con esto hemos copiado el link del repositorio.



6) Ahora vamos a nuestra consola nuevamente y digitamos el siguiente comando: git clone + link que hemos copiado anteriormente tener en cuenta los espacios al escribir el comando. De la siguiente manera:

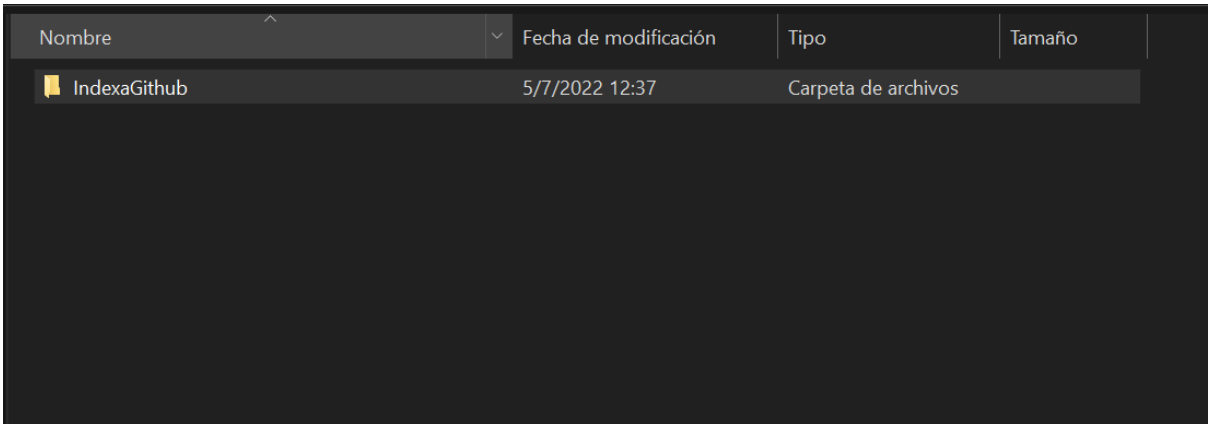


Luego presionamos enter y deberias ver lo siguiente:

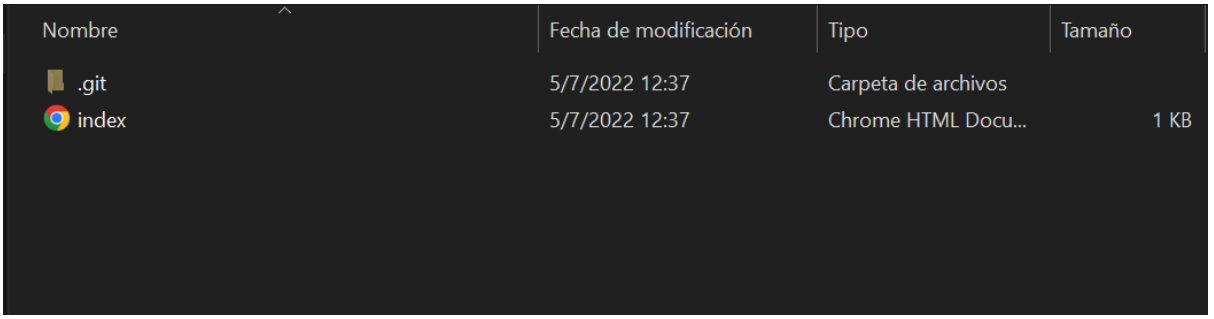


Esto nos indica que el repositorio se clonó correctamente.

7) Ahora debemos irnos a la carpeta donde abrimos la consola y debería aparecernos una carpeta que es donde está alojado todo el repositorio que hemos clonado. En el siguiente paso (8) veremos cómo entrar a la carpeta IndexaGithub o en sus casos la carpeta con su respectivo nombre, es importante hacerlo ya que debemos estar en esa carpeta para poder comenzar a trabajar.

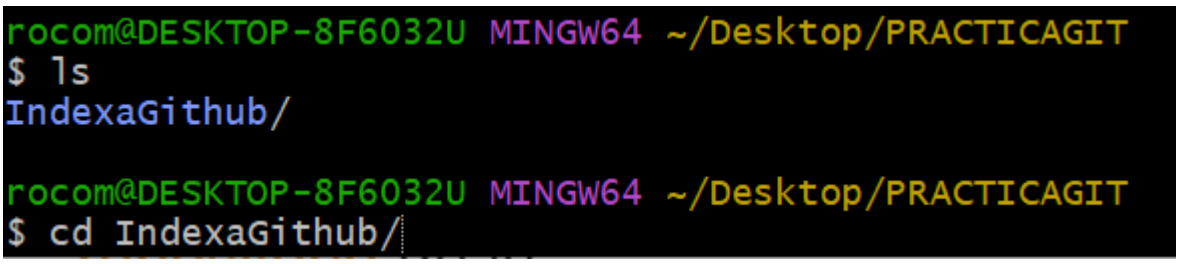


Si entramos en esa carpeta, en mi caso “IndexaGithub” veremos todos los archivos o carpetas que se encontraban en el repositorio que hemos clonado en mi caso veré lo siguiente una página index html ya que es lo único que contenía:

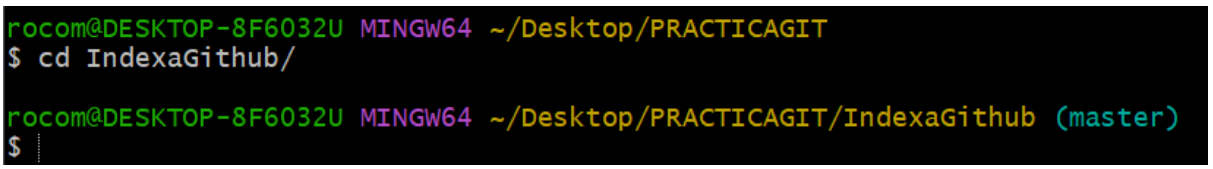


Si nos dirigimos al repositorio de Github veremos que se encuentran los mismos archivos. La carpeta .git no tenerla en cuenta, forma parte de los archivos de configuración que git crea automáticamente al clonar el repositorio. Si no la puedes ver es porque debes activar la opción para ver archivos ocultos.

8) Ahora digitamos el comando ls el cual nos mostrará la carpeta que se nos creó al clonar el repositorio, este comando lista todos los archivos que se encuentren en la carpeta en la que estamos. Luego de eso digitamos el siguiente comando cd + el nombre de la carpeta que se creó al clonar el repositorio/ De la siguiente manera y presionamos enter



Luego de presionar enter nos aparecerá lo siguiente:



Dónde podemos que nos aparece (master)

Lo que nos indica que estamos en la rama master del repositorio que hemos creado.
9)No es lo adecuado trabajar en la rama master ya que en esa rama es donde se encuentra la aplicación a desarrollar y no queremos romperla. Por lo tanto, debemos crear una nueva rama para trabajar ahí. Lo hacemos con el comando
git branch + nombre de nuestra rama

Escribimos eso y presionamos enter

Ahora digitamos el comando git branch y presionamos enter. Veremos un listado de las ramas.

Como pueden ver master está en color verde. Lo que nos indica que nos encontramos en esa rama.

10) Ahora debemos cambiarnos de la rama master hacia la rama que hemos creado. Lo hacemos de la con el siguiente comando:
git checkout + nombre de la rama

```
rocom@DESKTOP-8F6032U MINGW64 ~/Desktop/PRACTICAGIT/IndexaGithub (master)
$ git checkout developRodrigo
```

Y presionamos enter.. Veremos lo siguiente:

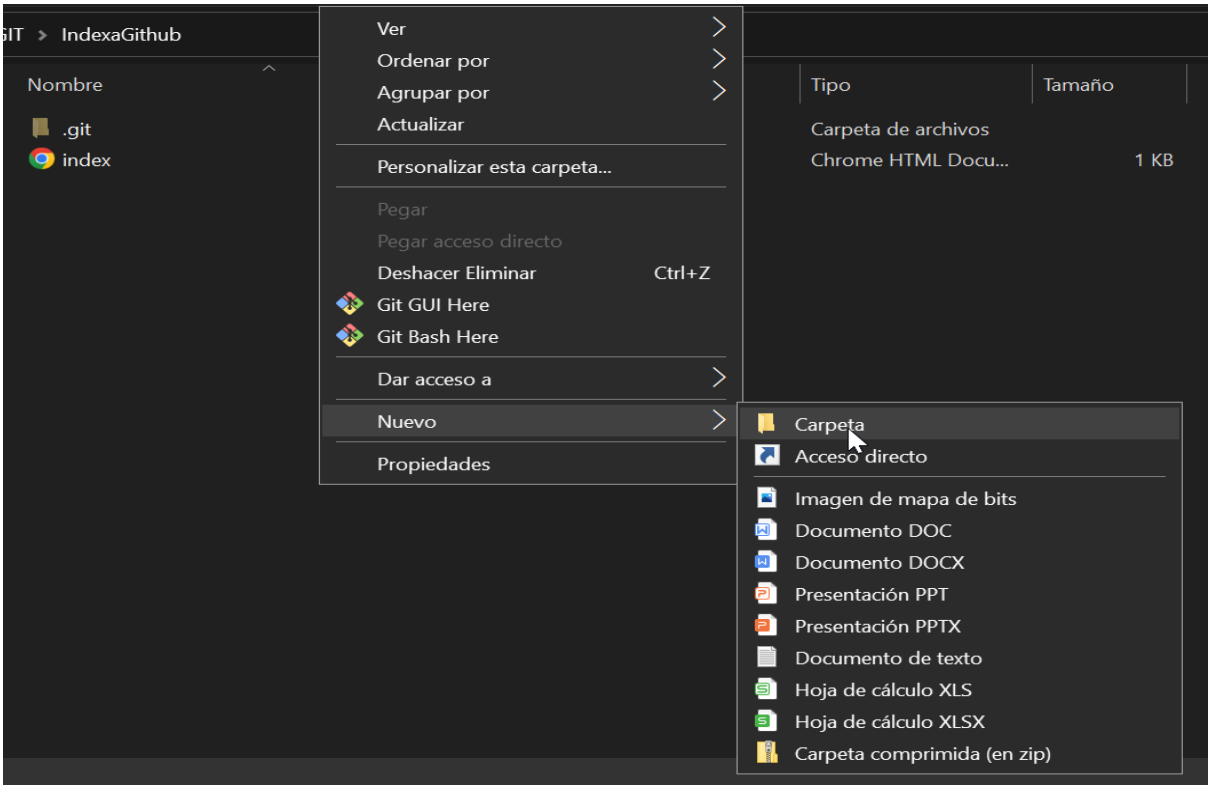
```
rocom@DESKTOP-8F6032U MINGW64 ~/Desktop/PRACTICAGIT/IndexaGithub (master)
$ git checkout developRodrigo
Switched to branch 'developRodrigo'

rocom@DESKTOP-8F6032U MINGW64 ~/Desktop/PRACTICAGIT/IndexaGithub (developRodrigo
)
$
```

Lo que nos indica que hemos movido correctamente a la rama que hemos creado. Como pueden ver en lugar de (master) me aparece (developRodrigo) lo que me indica que me encuentro en esa rama.

Si digitamos git branch y presionamos enter. Veremos que ahora developRodrigo aparece en color verde y la rama master en color blanco. Esto nos indica que nos hemos desplazado correctamente hacia la rama que hemos creado.

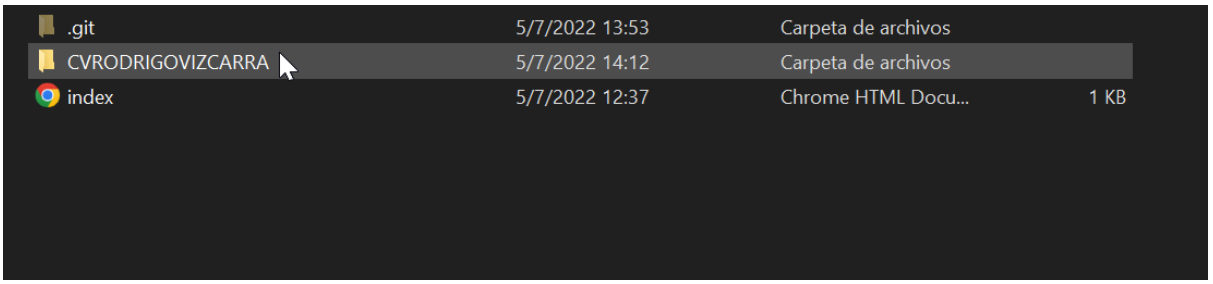
```
rocom@DESKTOP-8F6032U MINGW64 ~/Desktop/PRACTICAGIT/IndexaGithub (developRodrigo
)
$ git branch
* developRodrigo
  master

rocom@DESKTOP-8F6032U MINGW64 ~/Desktop/PRACTICAGIT/IndexaGithub (developRodrigo
)
$
```

11) Ahora debemos cargar nuestro CV en la carpeta que nos creó Git al clonar el repositorio. Volver al paso 8 para hacer memoria.

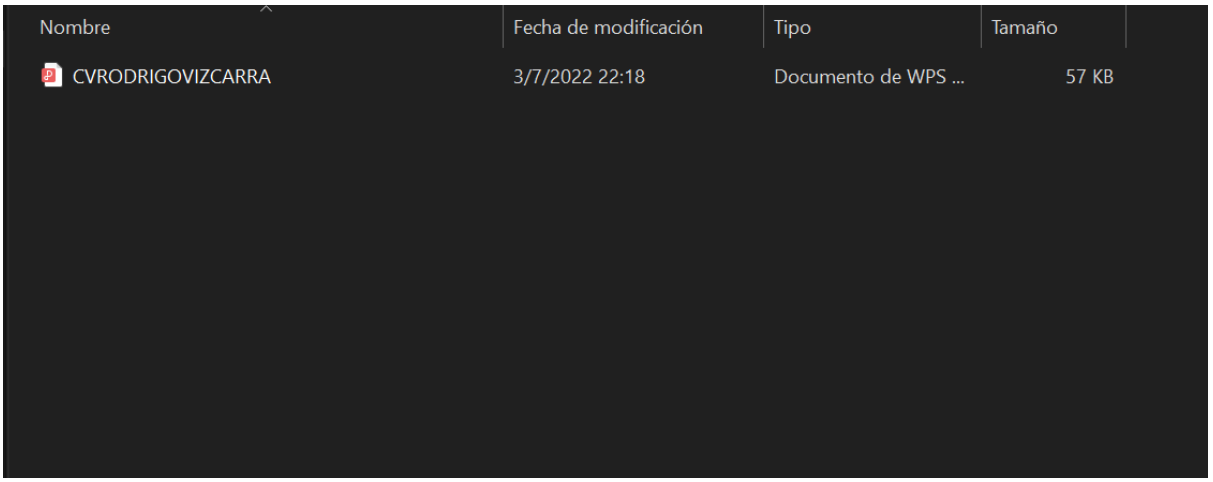


Creamos la carpeta y le pones el nombre que quieras. En esta carpeta es donde pondremos nuestro CV. Para proceder a subirla al repositorio de nuestro Grupo junto con nuestra rama propia.

Una vez creada procedemos a entrar en ella.

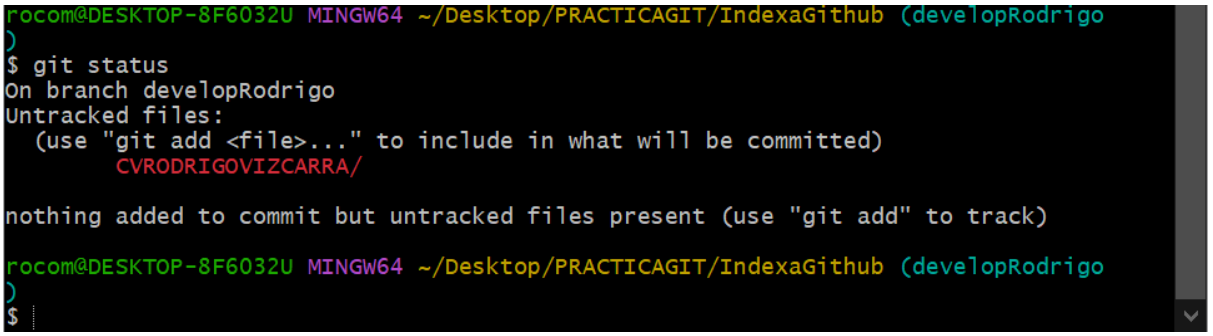


Ahora debemos ir a donde se encuentre nuestro CV, copiarlo y posteriormente pegarlo en la carpeta que hemos creado.



Como pueden ver, lo acabo de hacer.

12) Ahora volvemos a la consola de Git y digitamos el siguiente comando:
git status y presionamos enter. Les debería aparecer lo siguiente:

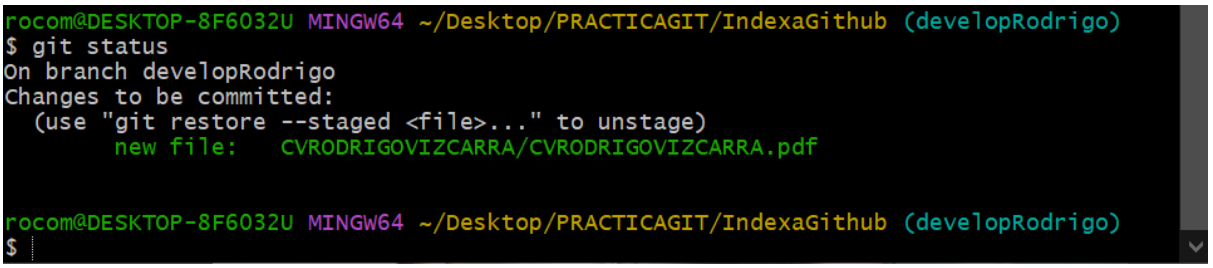


Nos aparece CVRODRIGOVIZCARRA/
(en sus casos les aparecerá el nombre que le hayan puesto a sus cv)

El color rojo indica que git reconoció el archivo pero debemos indicarle que debe darle un seguimiento. Lo hacemos con el siguiente comando:

git add + nombre del archivo en color rojo

Y presionamos enter. En principio no les aparecerá nada. Pero deben digitar git status y presionar enter nuevamente para ver si se añadió correctamente nuestro archivo. Deberían ver lo siguiente:



new file:

Esas líneas de new file en verde más el nombre de nuestro CV nos indica que se añadió correctamente nuestro archivo.

Pero aún hay otro paso más por hacer, muy muy importante. Que es el commit. Debemos hacer un commit de nuestro archivo. El commit es una confirmación más, por así decirlo, que debemos hacer. Esta confirmación se hace cuando estamos seguros de que nuestro código o el archivo que

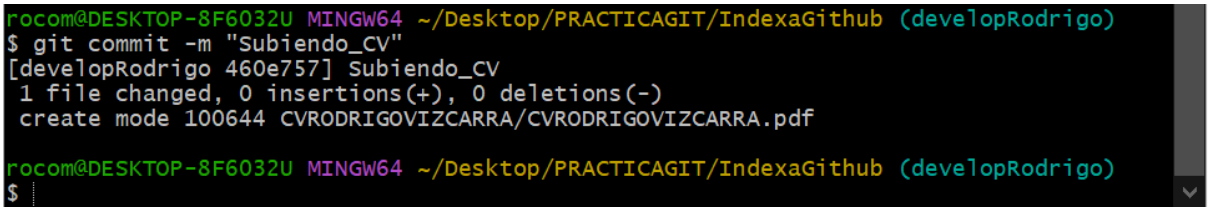
queremos modificar, eliminar, agregar, es el correcto. Lo veremos en el siguiente paso

Como pueden ver git status nos da el estado, nos muestra si hay un cambio o algún archivo nuevo que recientemente hayamos creado, modificado o traído de alguna parte y que debamos darle seguimiento con git add . Siempre digite el comando git status antes de cualquier otra cosa, para ver que todo esté en orden.

13) Ahora nos toca hacer un commit del archivo. Nuestro CV. Lo hacemos con el siguiente comando: git commit -m “comentario”
En comentario va un comentario que indique el cambio o que es lo que estamos por hacer commit.

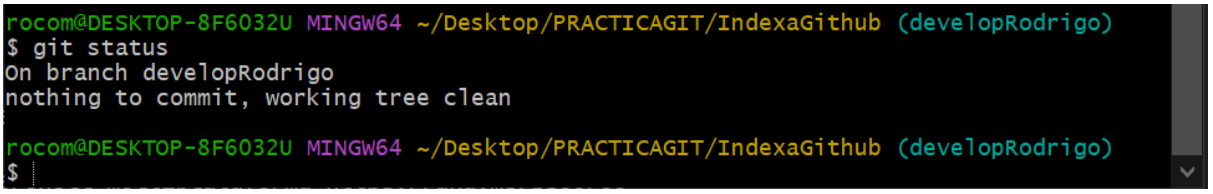
En mi caso el comentario que puse es “Subiendo_CV”
Escribimos ese comando que se ve en la captura y presionamos enter.

Deberias ver lo siguiente:



Nos indica que se hizo un commit correctamente de nuestro archivo. Y como pueden ver estamos dentro de nuestra propia rama. En mi caso esa rama es (developRodrigo) como pueden ver ahí.

Si digitamos git status y presionamos enter vamos a ver que ahora ya no nos aparece nada, nos dice que nos encontramos en la Rama developRodrigo y indica que ya no hay ningún cambio o archivo nuevo, etc que haya que añadir o commitear.



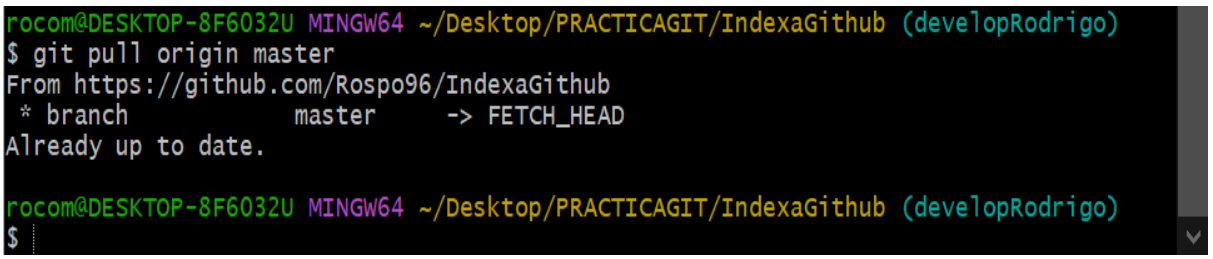
Si llegaste hasta aquí y no te ocurrió nada extraño y todo lo que mostré en las capturas te aparece exactamente como a mi, entonces hiciste todo bien FELICIDADES. Vamos al paso 14

14) Ahora debemos descargar cualquier posible cambio que haya en el repositorio remoto en Github. Lo hacemos con el siguiente comando:
git pull origin master

git pull origin main

IMPORTANTE ANTES DE DIGITAR EL COMANDO
Dependiendo de si tu rama principal se llama “master” o “main” tu pondrás main en el comando anterior en lugar de master. Yo pongo master ya que mi rama principal se llama master. Si tu rama principal se llama main, debes poner main.
Esto actualizará nuestra rama master o main para que descargue cualquier cambio que haya en la rama main o master del repositorio remoto. Lo cual mantendrá actualizada nuestra rama principal main o master.

En mi caso digitare “git pull origin master”
Presionamos enter y veremos lo siguiente:



Lo que nos indica que hemos actualizado los últimos cambios realizados en la rama master o main del repositorio remoto y los hemos descargado hacia nuestra rama main o master de nuestro repositorio local.

git pull es importante tener en cuenta ya que si queremos subir cambios, sin antes haber descargado cualquier otro cambio que nuestros compañeros hayan subido, producirá errores que no queremos.

15) Ahora procedemos a subir nuestra rama que hemos creado junto con la carpeta donde esta nuestro CV
Procedemos a digitar el siguiente comando:
git push origin master

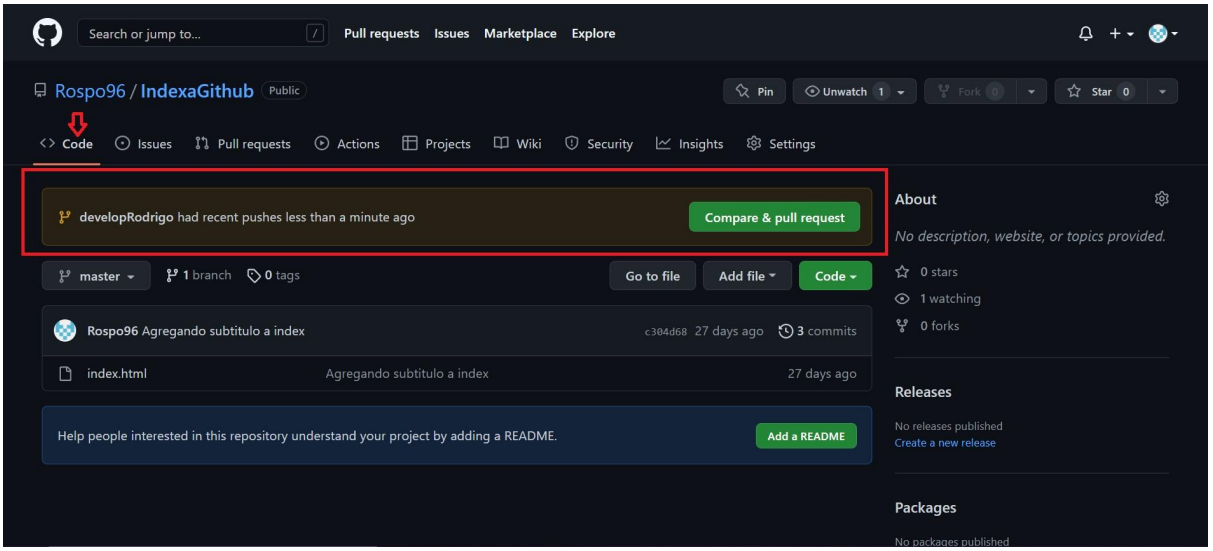
git push origin main

Pondrás master o main dependiendo de tu caso.
Presionas enter y verás o deberías lo siguiente:

```
rocom@DESKTOP-8F6032U MINGW64 ~/Desktop/PRACTICAGIT/IndexaGithub (developRodrigo)
$ git push origin developRodrigo
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 54.79 KiB | 10.96 MiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'developRodrigo' on GitHub by visiting:
remote:      https://github.com/Rospo96/IndexaGithub/pull/new/developRodrigo
remote:
To https://github.com/Rospo96/IndexaGithub.git
 * [new branch]      developRodrigo -> developRodrigo

rocom@DESKTOP-8F6032U MINGW64 ~/Desktop/PRACTICAGIT/IndexaGithub (developRodrigo)
$
```

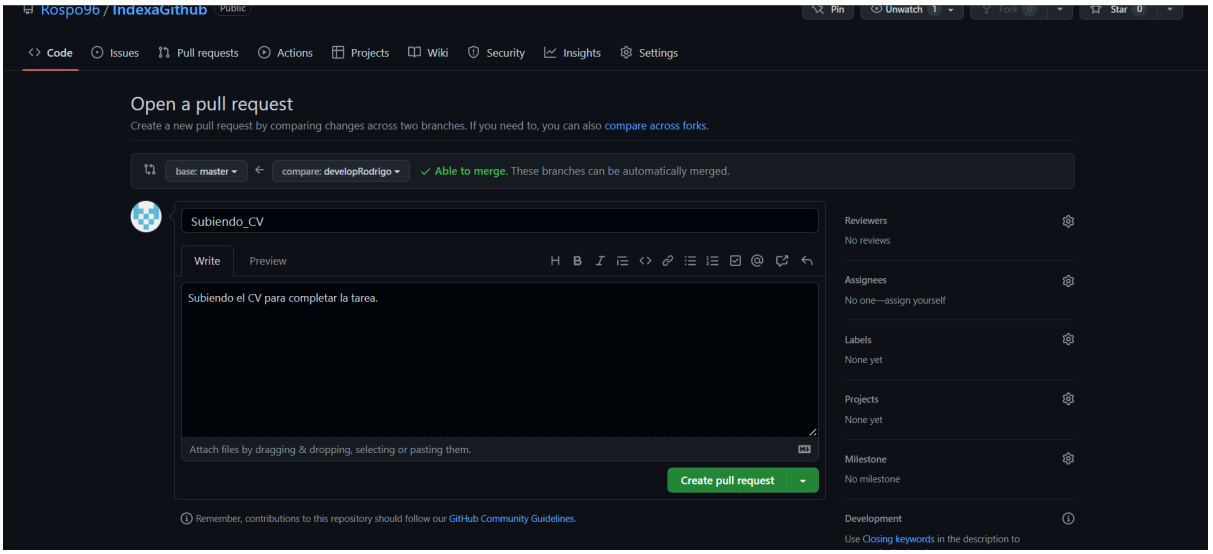
Lo que nos indica que hemos subido correctamente nuestra rama junto con la carpeta que hemos creado y el cv que hemos puesto en ella al repositorio remoto.

16) Ahora debemos ir al repositorio de github y veremos lo siguiente:



Esto nos indica el nombre de nuestra rama, la que creamos.
Y el botón verde “Compare & pull request” Comparar y solicitar extracción.

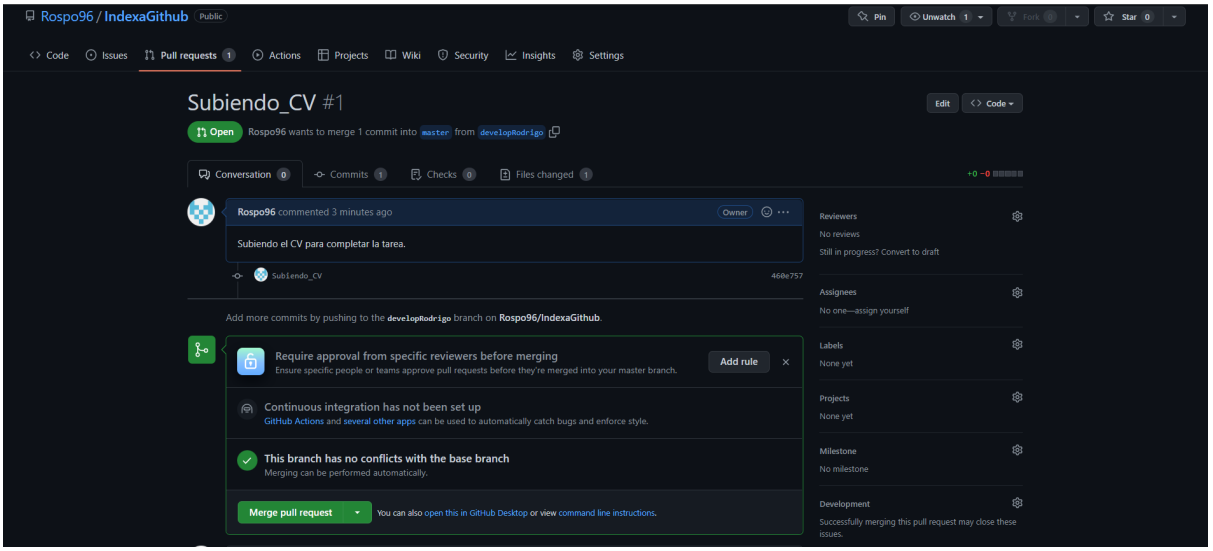
Al presionar el botón verde “Compare & pull request” veremos la siguiente pantalla:



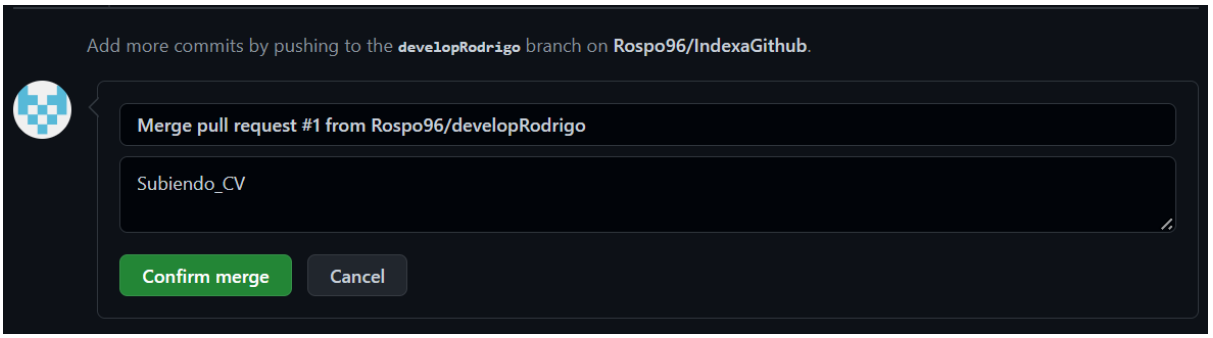
Escribimos un comentario en este caso “Subiendo el CV para completar la tarea”. Y pulsamos en el botón verde “Create pull request”

Lo cual nos redireccionará a la siguiente ventana: Proporcionándonos información de la fusión de los cambios y que la rama no cuenta con conflictos. Esto gracias a que antes de hacer push hicimos el pull que nos descargó los últimos cambios que se encontraban en la rama base main o master según sea el caso.

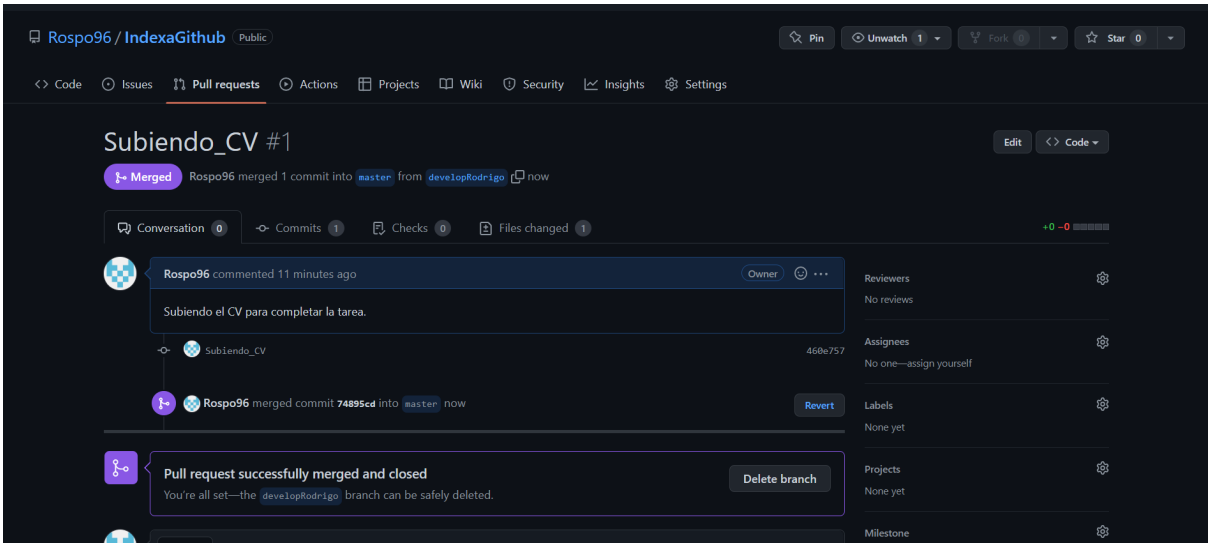
En esta pantalla debemos presionar el botón verde “Merge pull request” que está más abajo



Nos aparecerá la siguiente pantalla en la que debemos hacer click en el botón verde “Confirm merge”



Ahora deberíamos ver la siguiente pantalla:



“Pull request successfully merged and closed”
Lo que nos indica que la fusión fue correcta.

Si nos vamos a <> Code y luego a la rama master o main. Veremos que efectivamente se subieron los cambios a la rama main.