

# Universidad de las fuerzas armadas “ESPE”

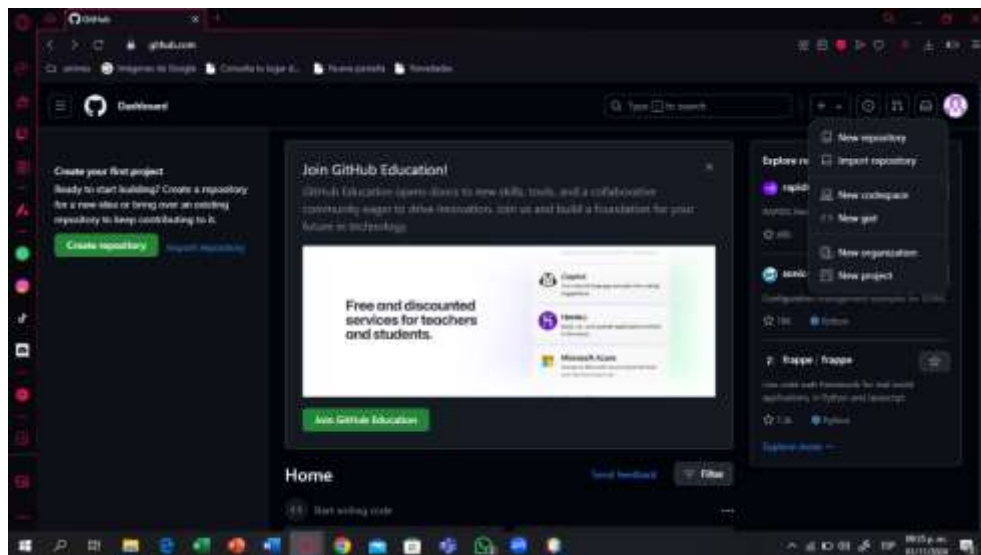
Nombre: David Calvopiña

NRC: 1940

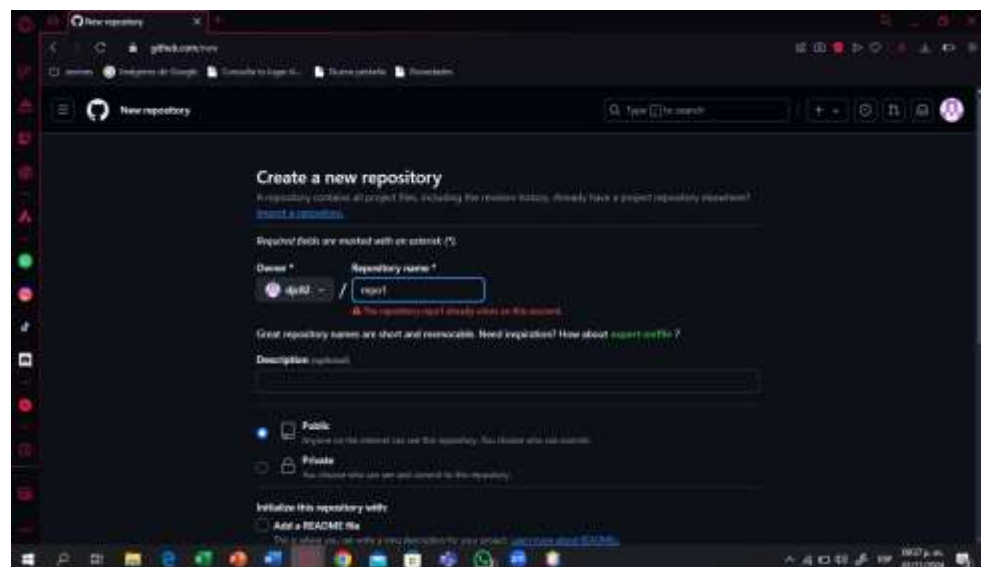
FECHA: 01/11/2024

## Pasos para crear un repositorio

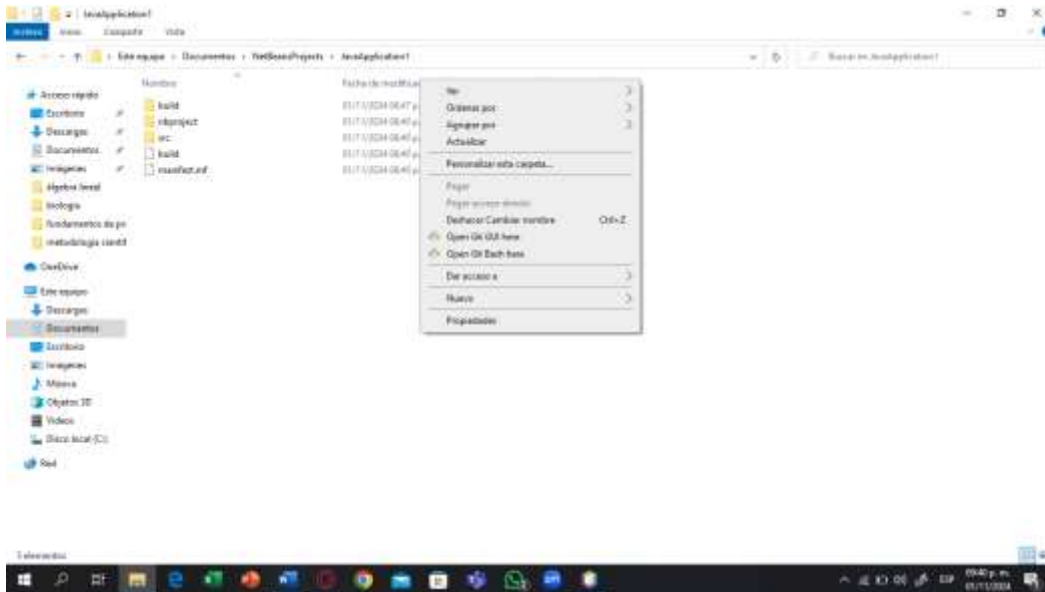
- 1) Primero crear una cuenta en github, luego dar click en **crear repositorio**



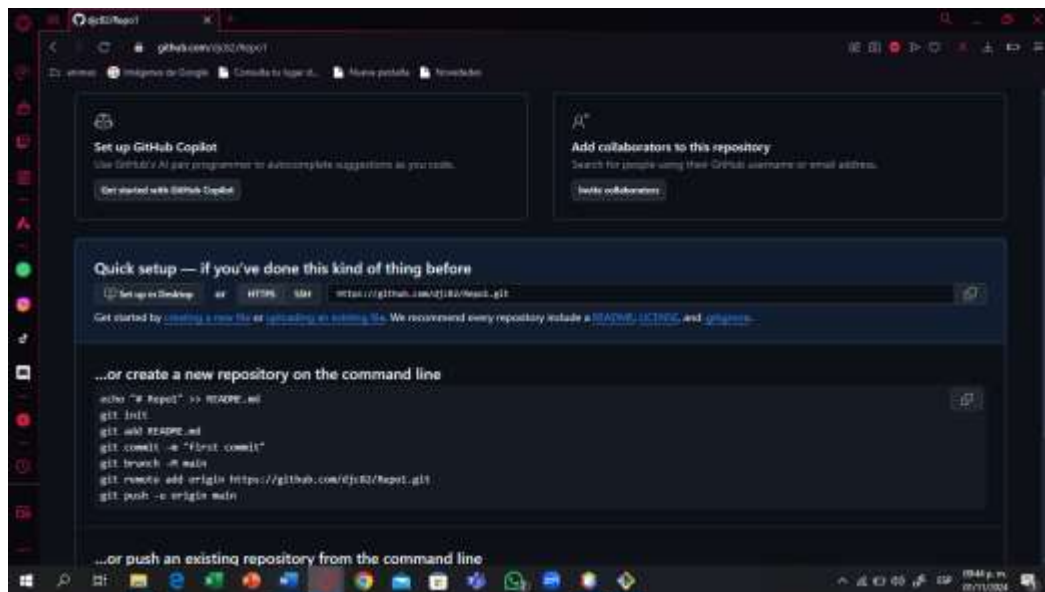
- 2) Luego colocar el nombre del repositorio y seleccionar si el repositorio va hacer privado o publico



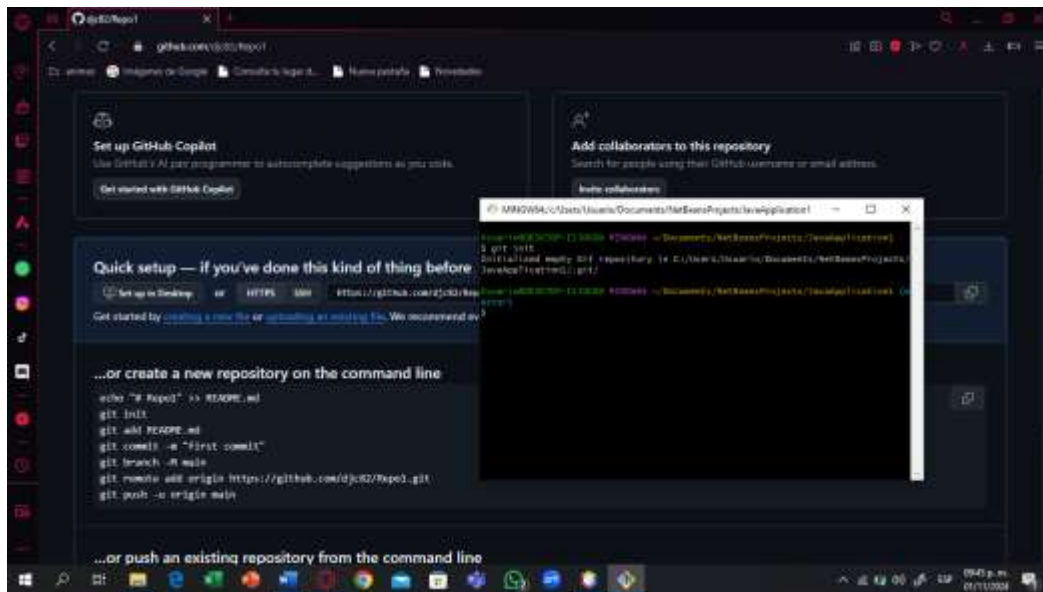
- 3) Entrar a la carpeta en donde se encuentra el proyecto y dar click a Gitbash para poder iniciar el repositorio.



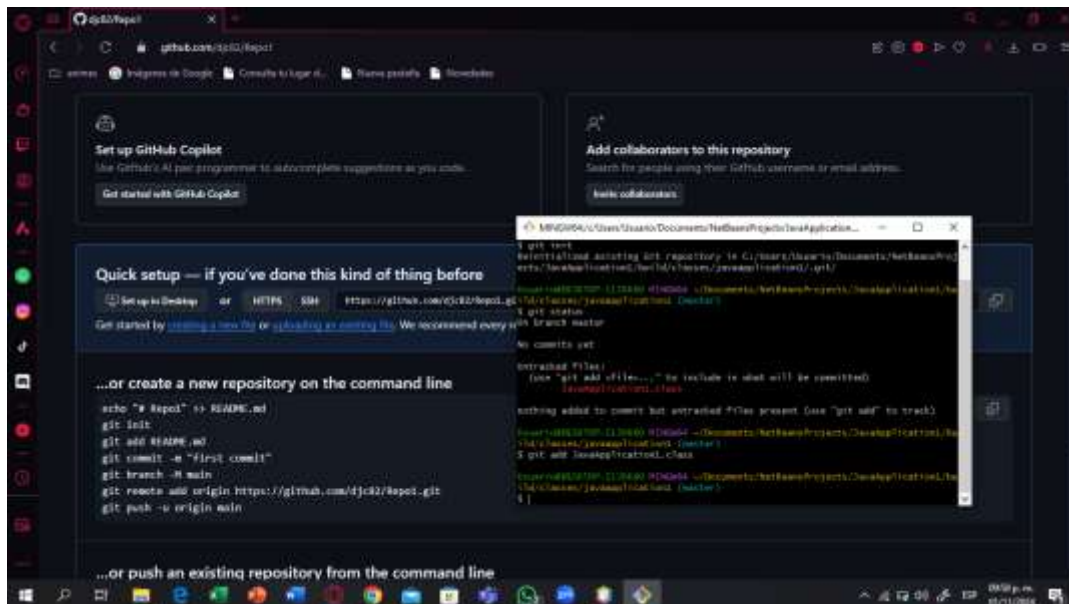
- 4) Volver a github y seguir los siguientes pasos.



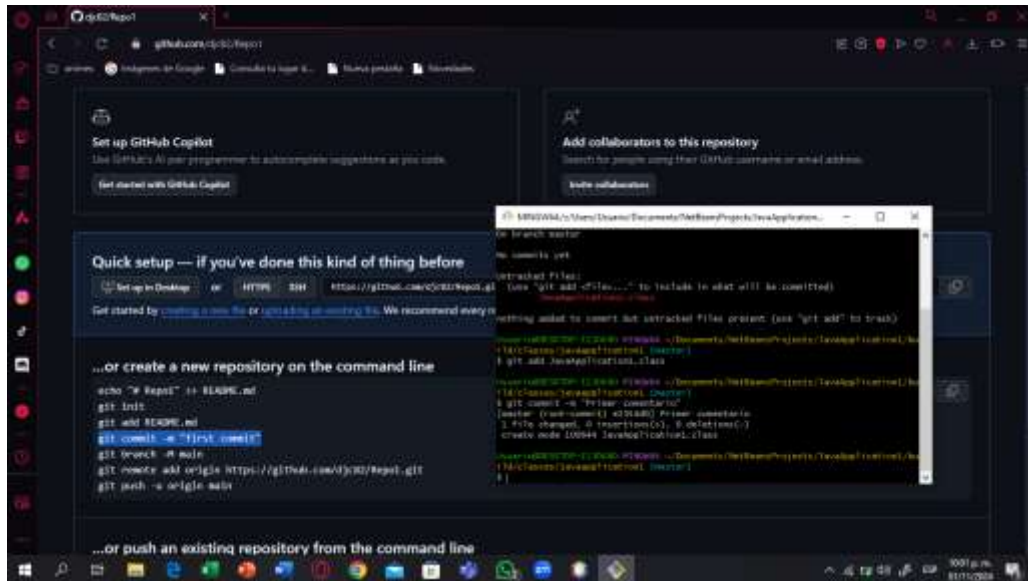
5) En gitbash colocar “git init”



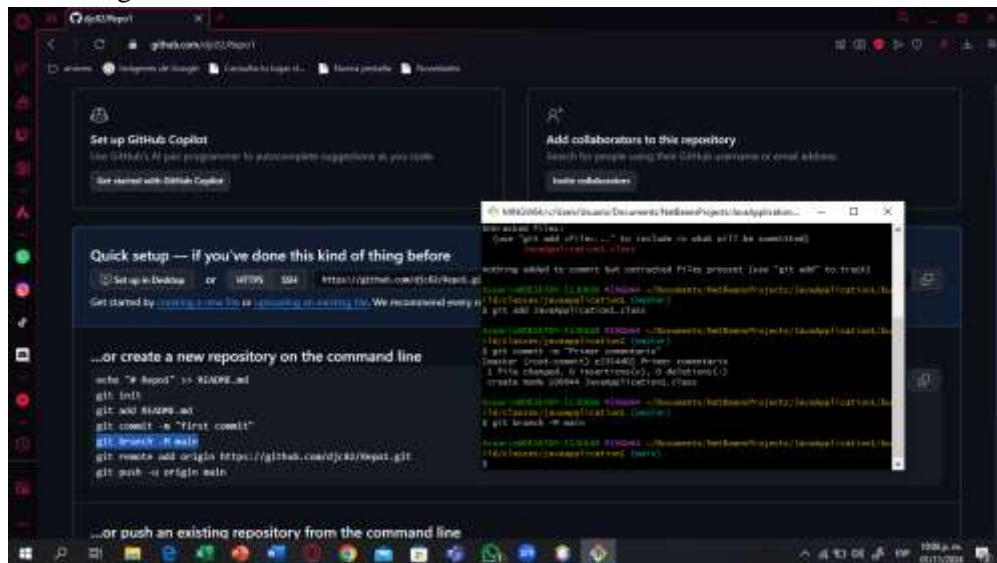
6) Luego colocar “git add” y si se quiere subir un archivo en específico se coloca el nombre del archivo.



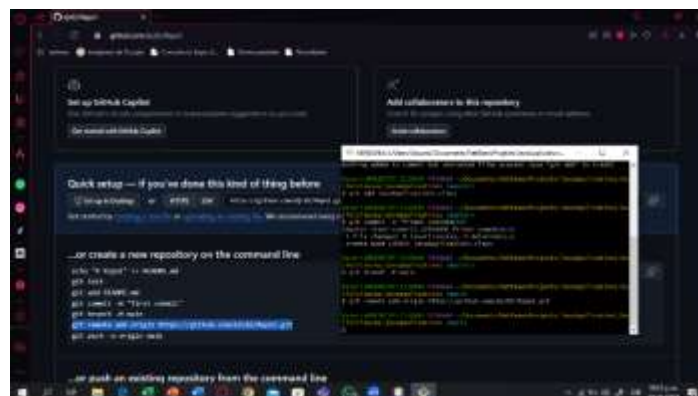
7) Colocar "git commit -m "first commit""



8) Colocar "git branch -M main"



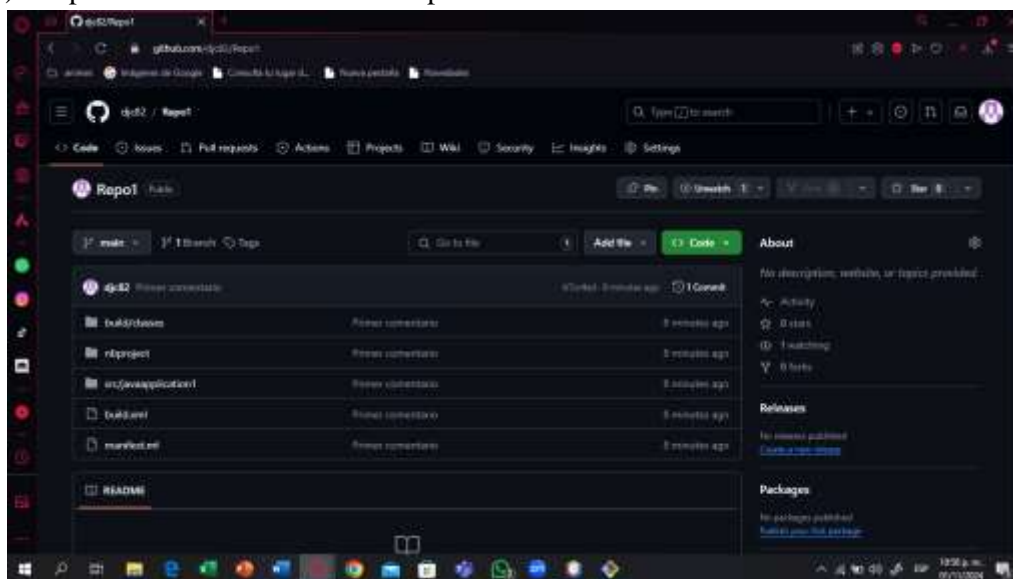
9) Subir al repositorio remoto



10) Agregar “git push -u origin main”

[illegible]

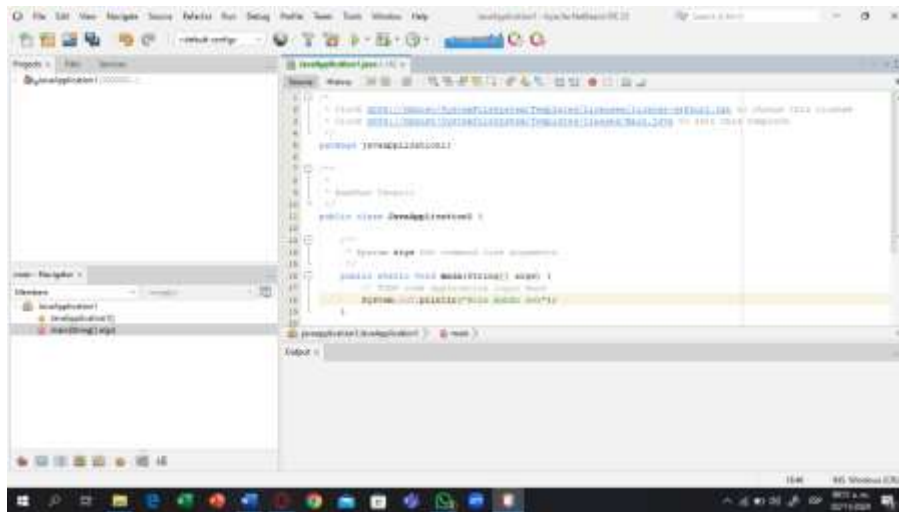
11) Despues de esto se ha creado el repositorio



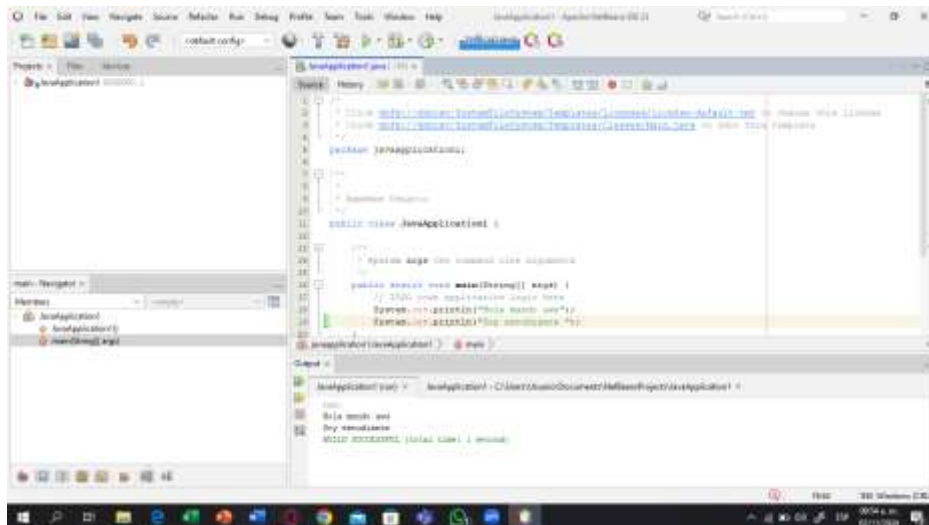
## Modificación de código

12) Una vez creado el repositorio se modifica el código Original

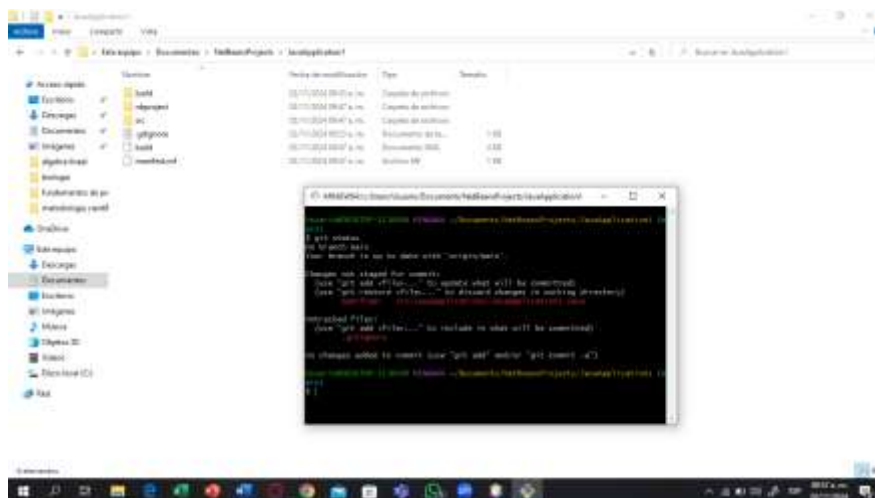




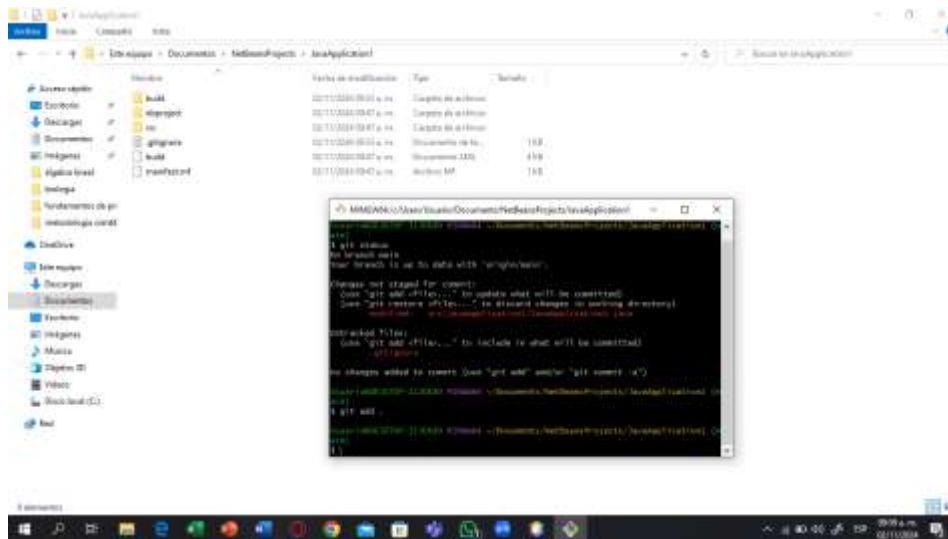
## Modificación



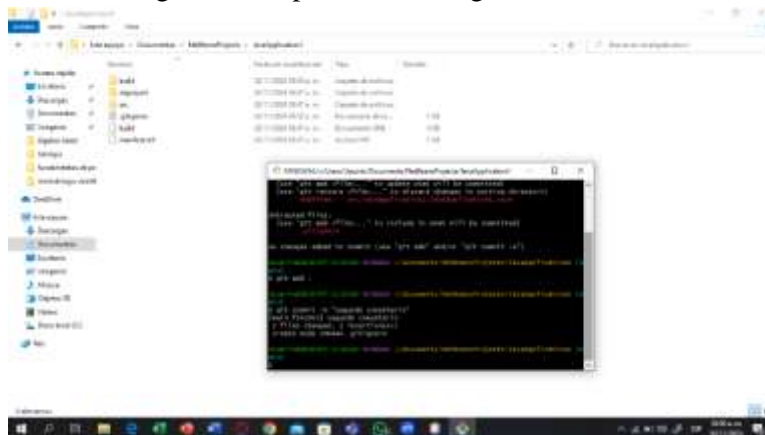
13) Una vez modificado el código, nos dirigimos al archivo y volvemos abrir git bash para poder subir el segundo comentario



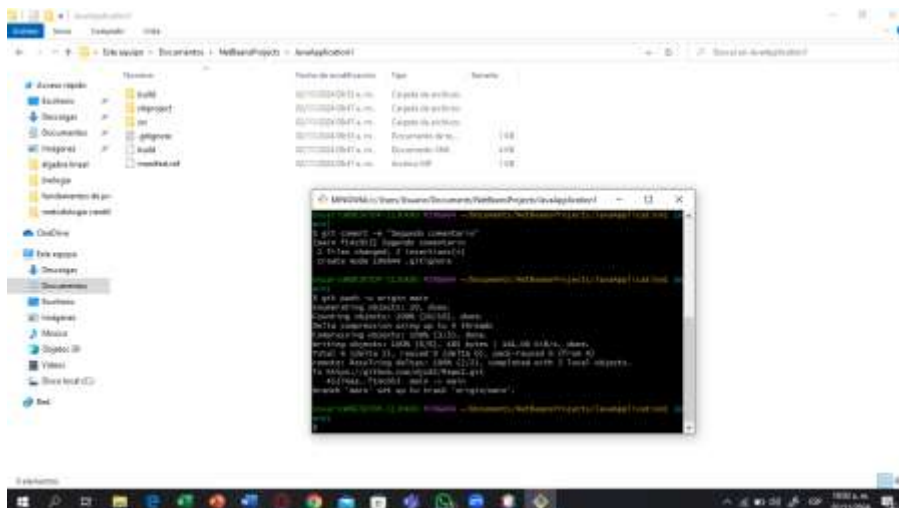
14) En el git bash colocamos git add .

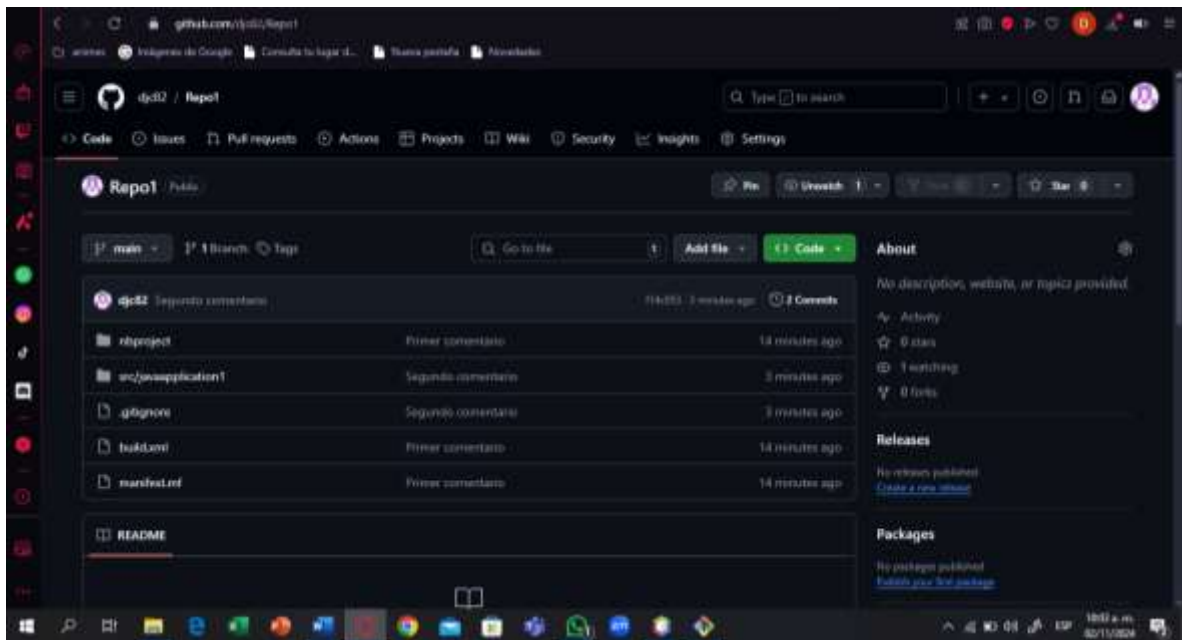


15) De ahí colocamos un git commit para crear el segundo comentario



16) Por ultimo colocamos git push -u origin main para que se guarde el nuevo repositorio el github





Recuperación de una versión anterior

17) Para ver cuantas modificaciones se han realizado colocamos git log en git bash

```

C:\Users\Usuario\Documents\NetBeansProjects\JavaApplication1
git log
commit 1a094532d54678734677855118a3e0d3d (HEAD -> main, origin/main)
Author: djc82 <djcalvo@unah.edu.ec>
Date: Sat Nov 2 10:05:12 2024 -0500

    Tercer comentario

commit f4c203d37a6a6a1279a00065d223a12e6d
Author: djc82 <djcalvo@unah.edu.ec>
Date: Sat Nov 2 10:00:12 2024 -0500

    Segundo comentario

commit 21274a677967c2b10d0114b749f3c20e4b
Author: djc82 <djcalvo@unah.edu.ec>
Date: Sat Nov 2 09:49:13 2024 -0500

    Primer comentario
  
```

18) Para regresar a la primera versión del código, se copia el comentario en amarillo y se coloca git checkout commit amarillo



```
MINGW64 ~/Users/Usuario/Documents/NetBeansProjects/JavaApplication1
$ git checkout HEAD --> main, origin/main
commit 34d4652210d40b75744577865e13b3e0c28 (HEAD --> main, origin/main)
Author: djc82 <djc82@calvopi.net>
Date: Sat Nov 2 10:01:11 2024 -0500

    Tercer comentario

commit 74c2026376a306179e00b05e0c311a32e0d
Author: djc82 <djc82@calvopi.net>
Date: Sat Nov 2 10:00:32 2024 -0500

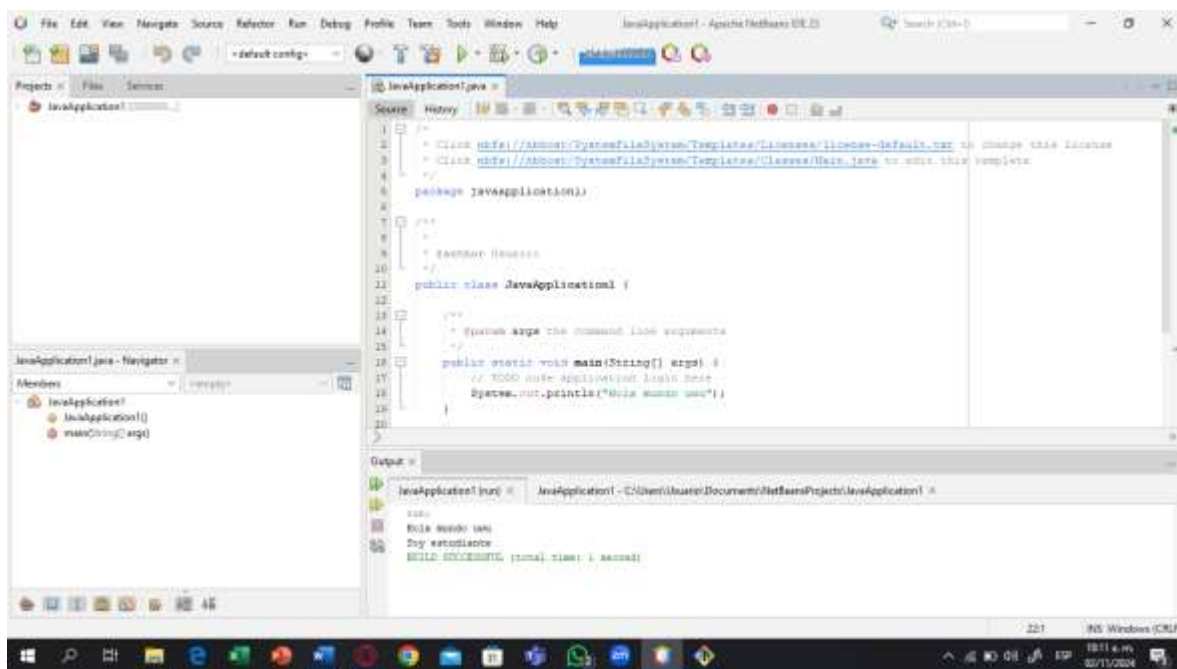
    Segundo comentario

commit 43274ae7f067fc21b0231e0f48fccc9bd5
Author: djc82 <djc82@calvopi.net>
Date: Sat Nov 2 09:49:11 2024 -0500

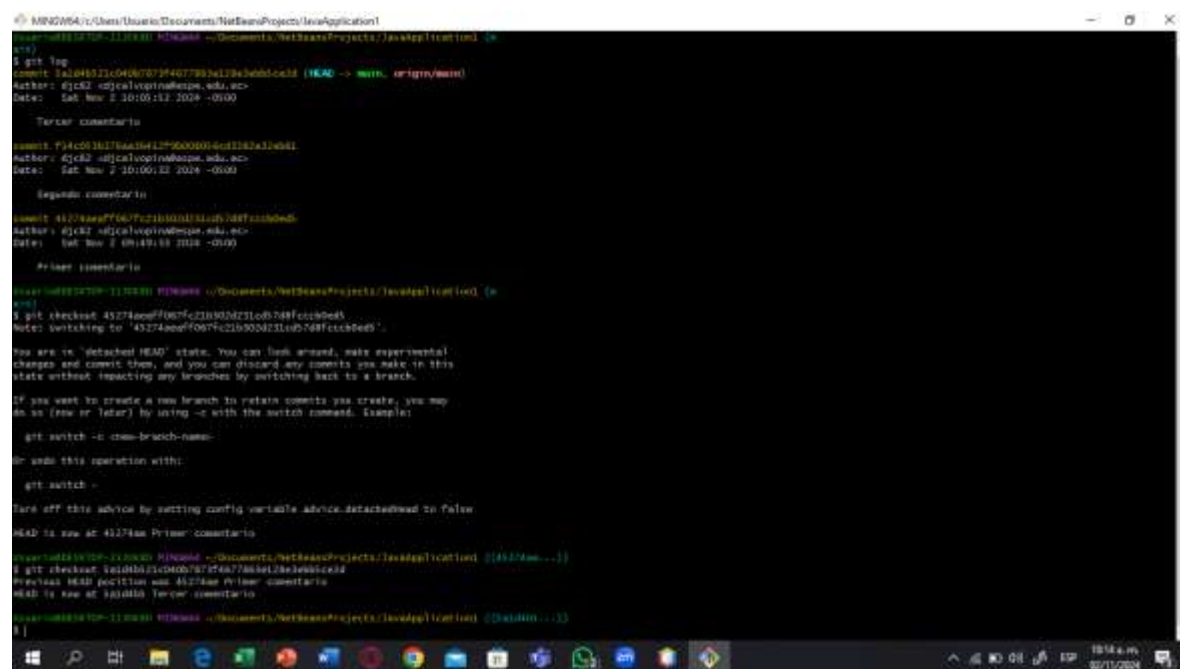
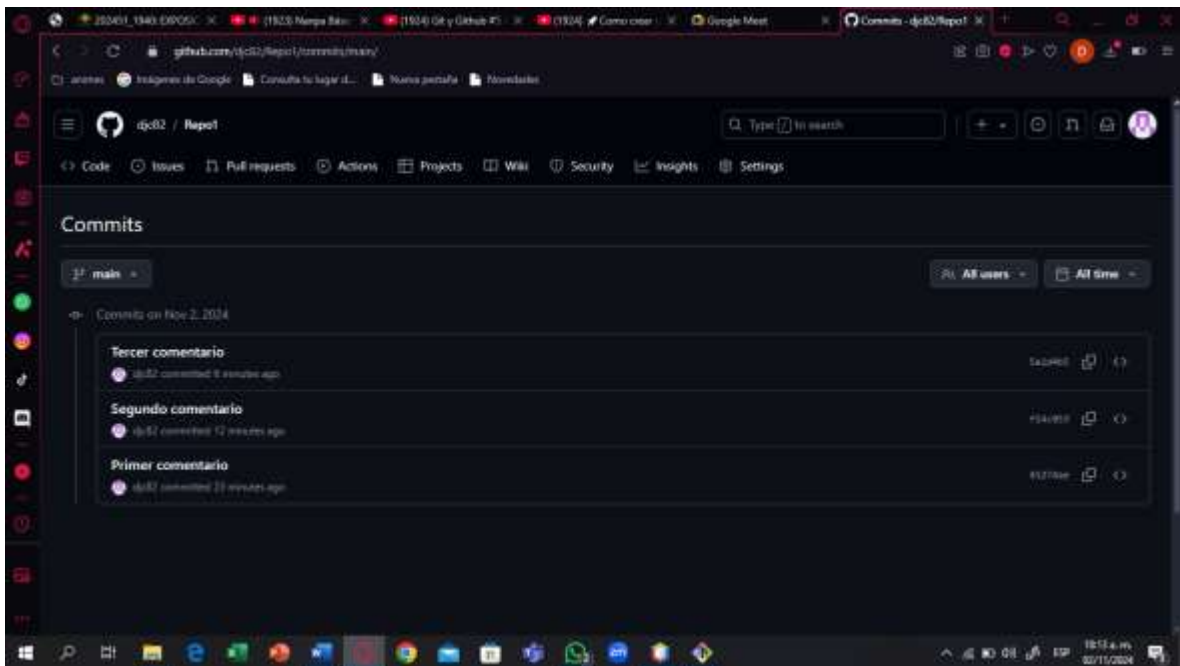
    Primer comentario

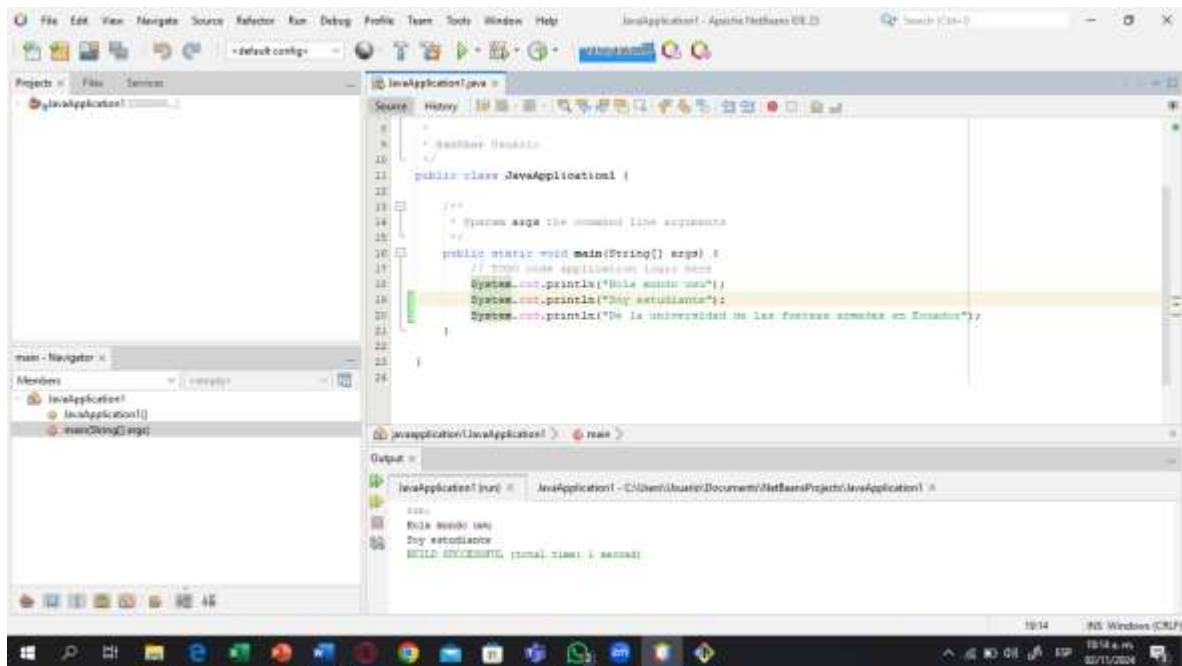
MINGW64 ~/Users/Usuario/Documents/NetBeansProjects/JavaApplication1 $
$ git checkout HEAD --> main, origin/main
Note: switching to '43274ae7f067fc21b0231e0f48fccc9bd5'.
You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.
If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:
  git switch -c <new-branch-name>
Or undo this operation with:
  git switch -
Turn off this advice by setting config variable advice.detachedHead to false
HEAD is now at 43274ae Primer comentario
MINGW64 ~/Users/Usuario/Documents/NetBeansProjects/JavaApplication1 ([43274ae...])
$
```

De ahí el código ha vuelto a su primera versión



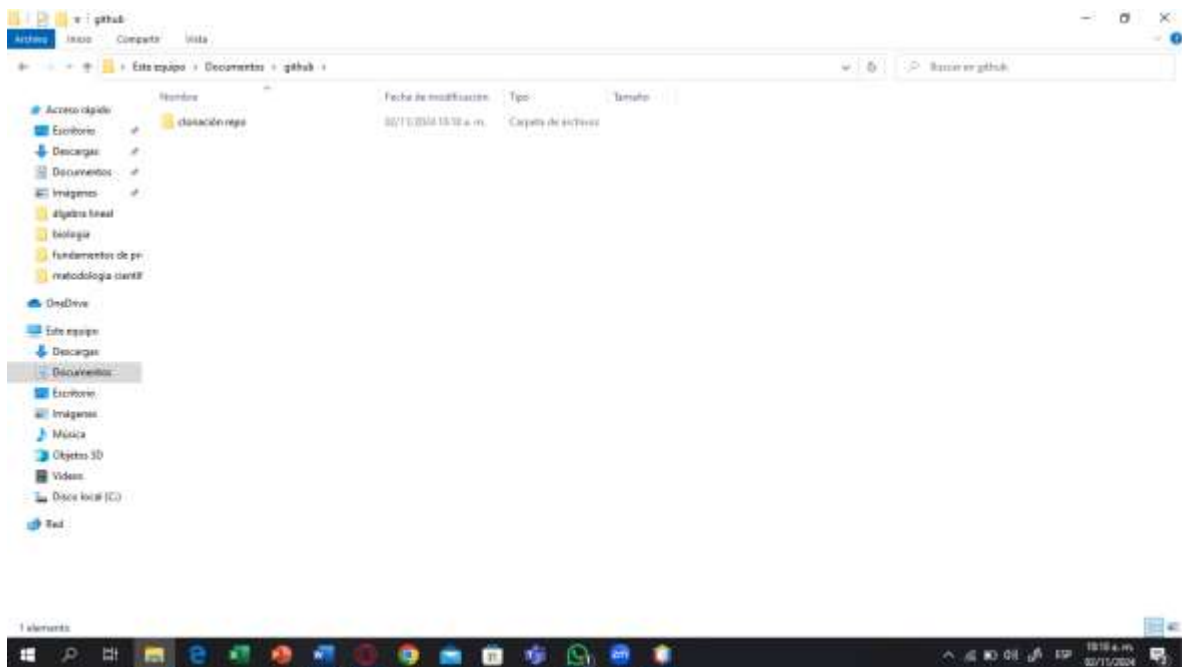
19) Para ir a la última versión nos dirigimos a commits en github y se copia el comentario de la versión que se requiera



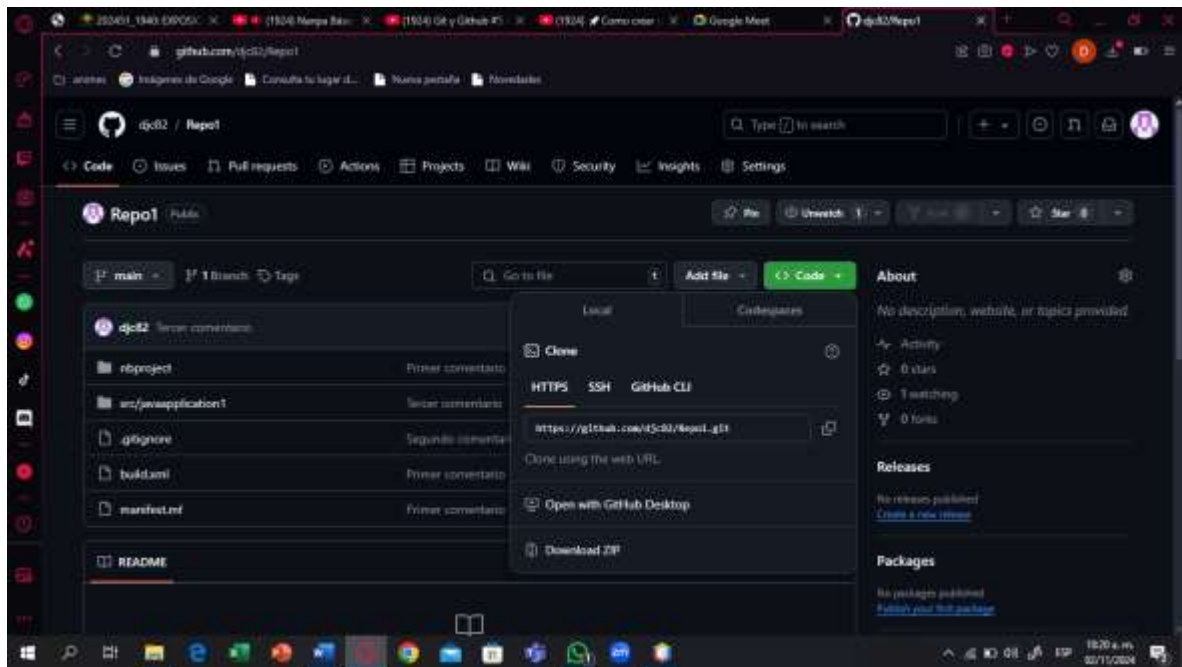


Clonar repositorio

20) Creamos una nueva carpeta, en donde queramos que este la copia



21) Después nos dirigimos a github, a la parte de code y copiamos el url



22) Una vez hecho eso nos dirigimos a la carpeta y en git bash colocamos git clone y el url .

