Seminario GitHub

2019

Rocío Utrera Rodriguez

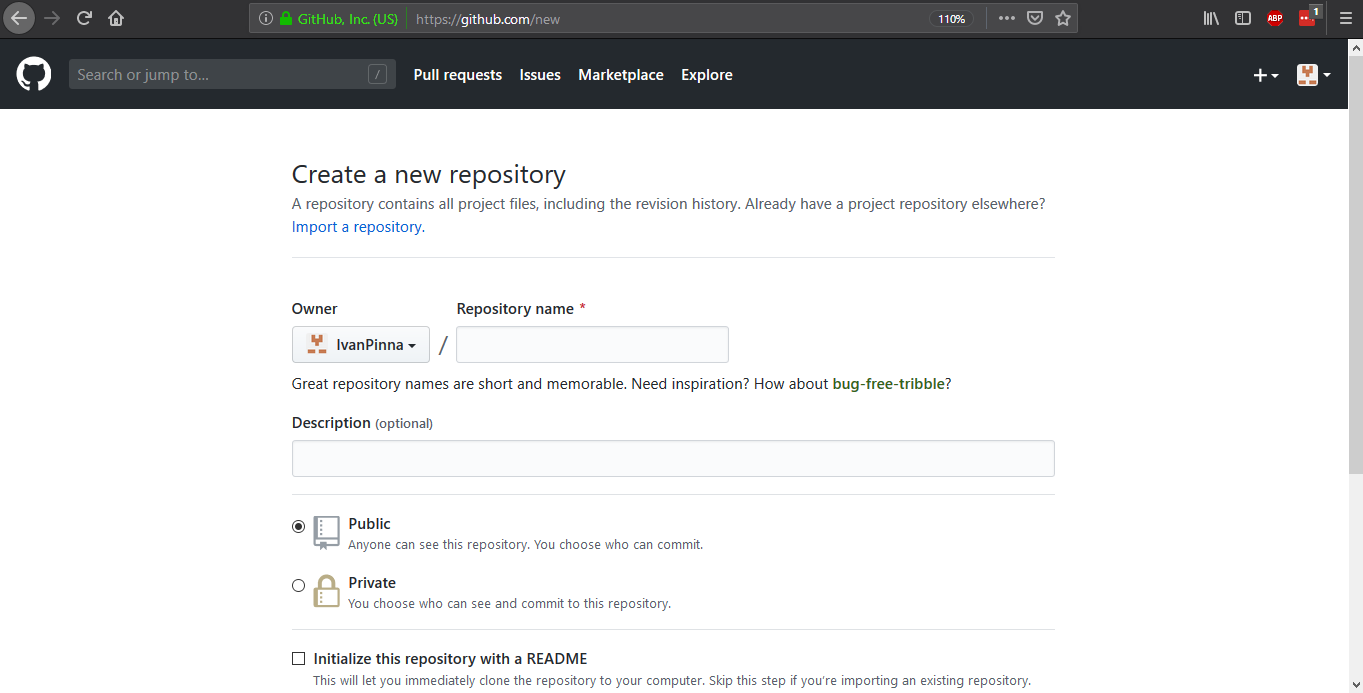
Iván Piña Arévalo

Moises Guerrero López

Escuela superior de ingeniería, Universidad de cádiz | 21/05/19

En esta actividad explicaremos como subir un archivo a GitHub así como los comandos necesarios para sincronizar el repositorio, subir archivos y modificarlos.

El encargado de crear el repositorio fue Iván. Iván lo realizó a través de la página web de Github, con la opción “create new repository”



Después, lo único necesario fue abrir una terminal y teclear

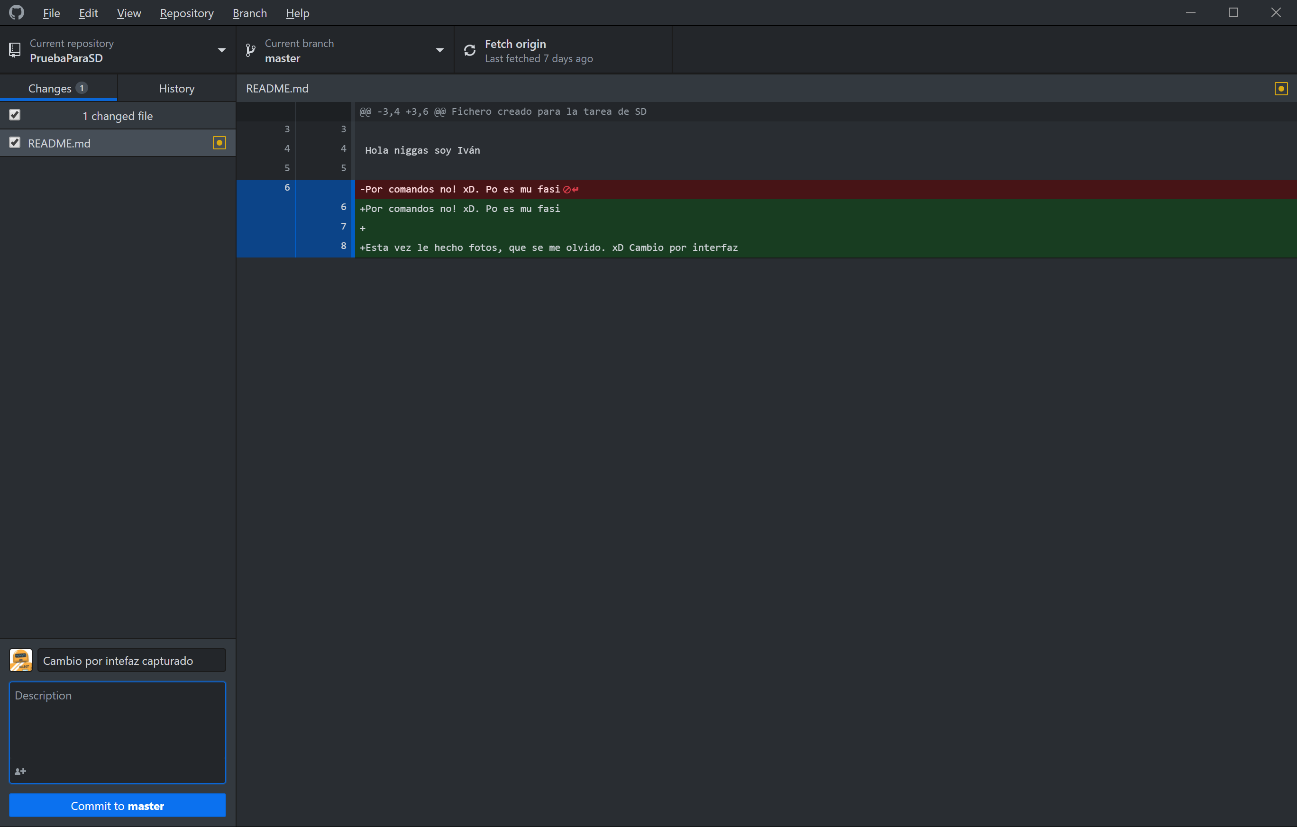
“git clone <https://github.com/IvanPinna/PruebaParaSD>”

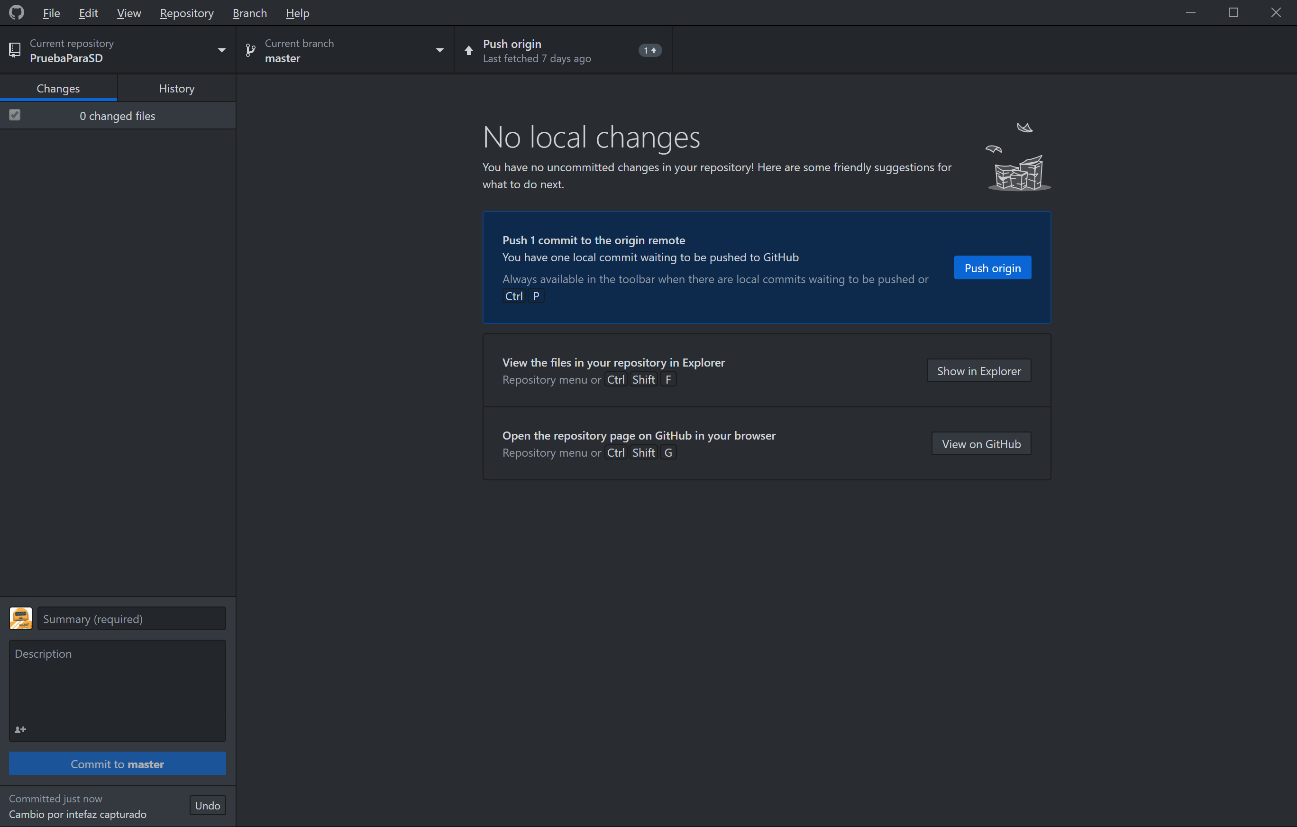
De esta manera se replica el repositorio en el ordenador.

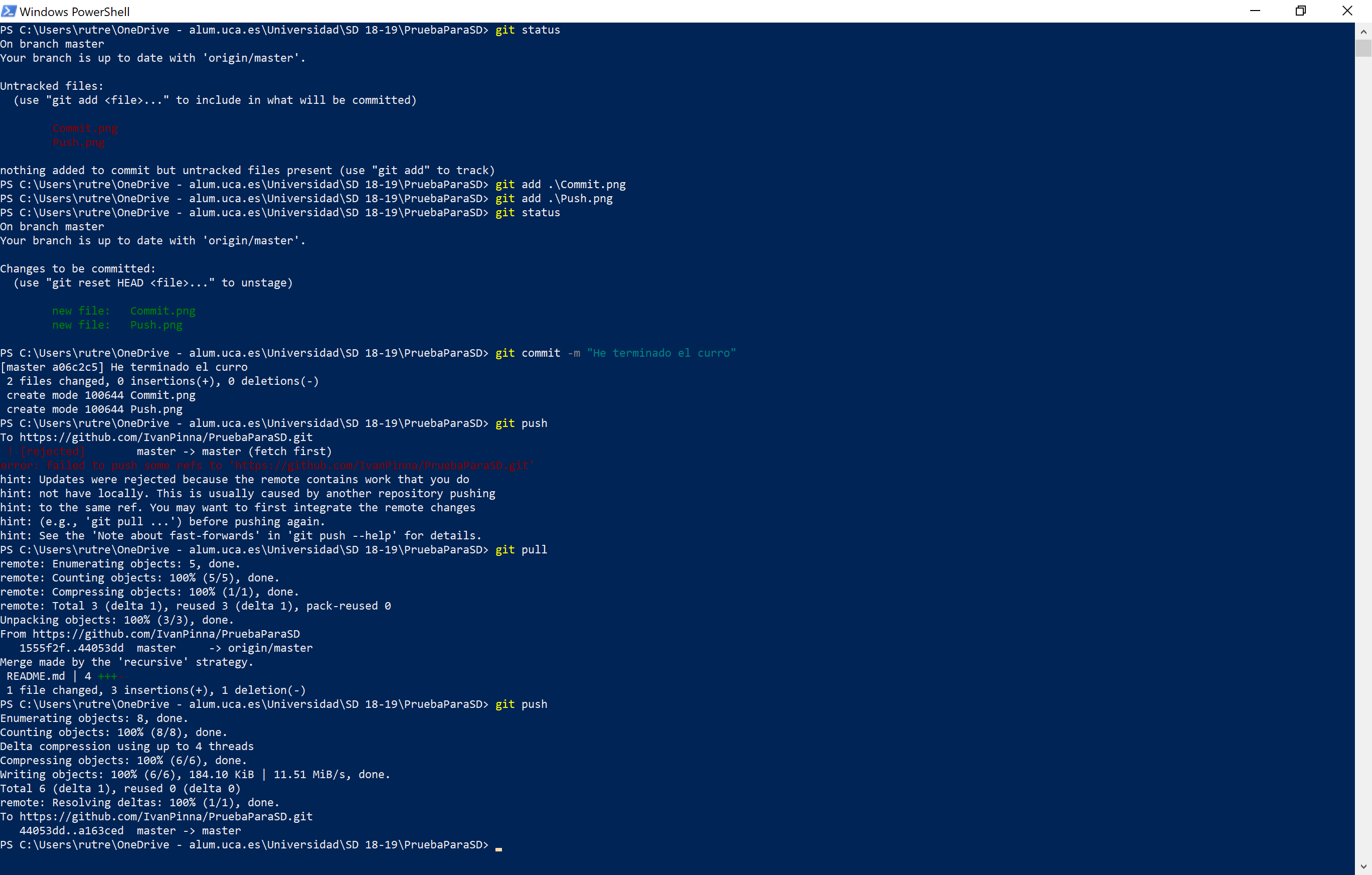
Por una parte, Rocío subió archivos con la interfaz gráfica y por comandos. Se explicará primero por la interfaz gráfica.

Una vez instalado la interfaz y creado nuestro repositorio, para añadir algún archivo o modificarlo, solo hay que añadir o modificar en la carpeta local. Luego darle a commit. Y finalmente pulsamos el botón push.

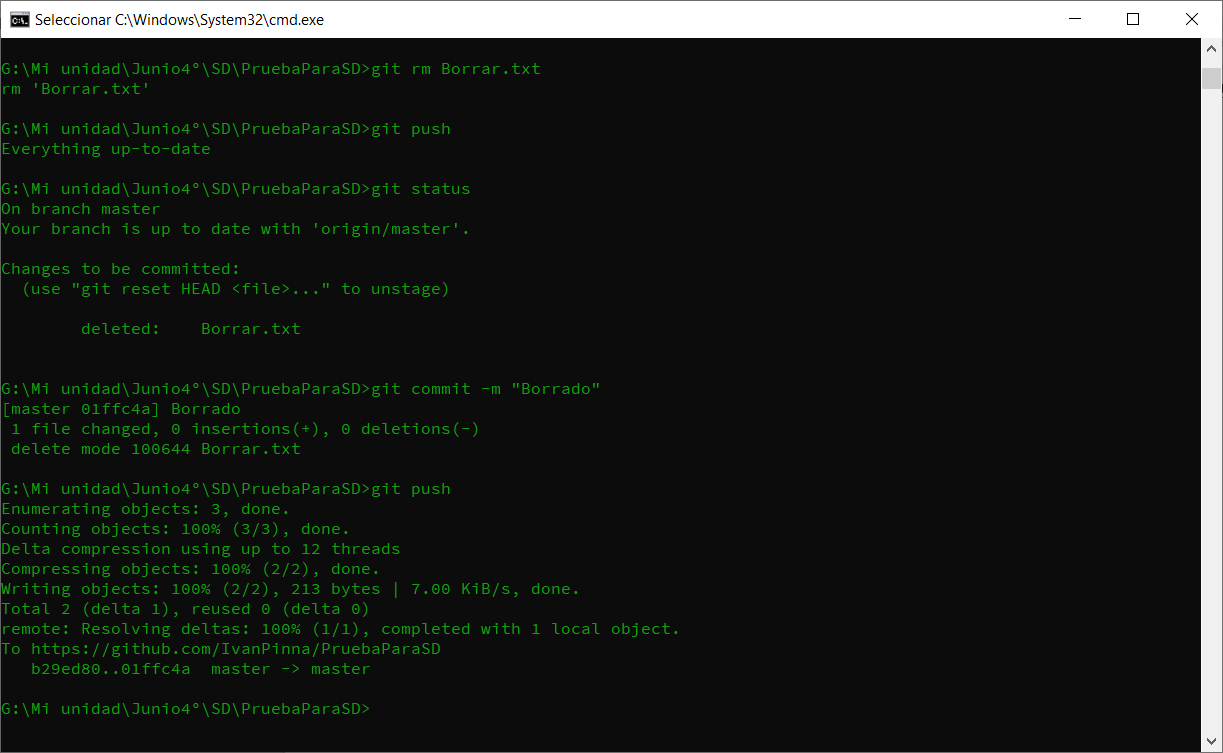
Mostramos las imágenes.





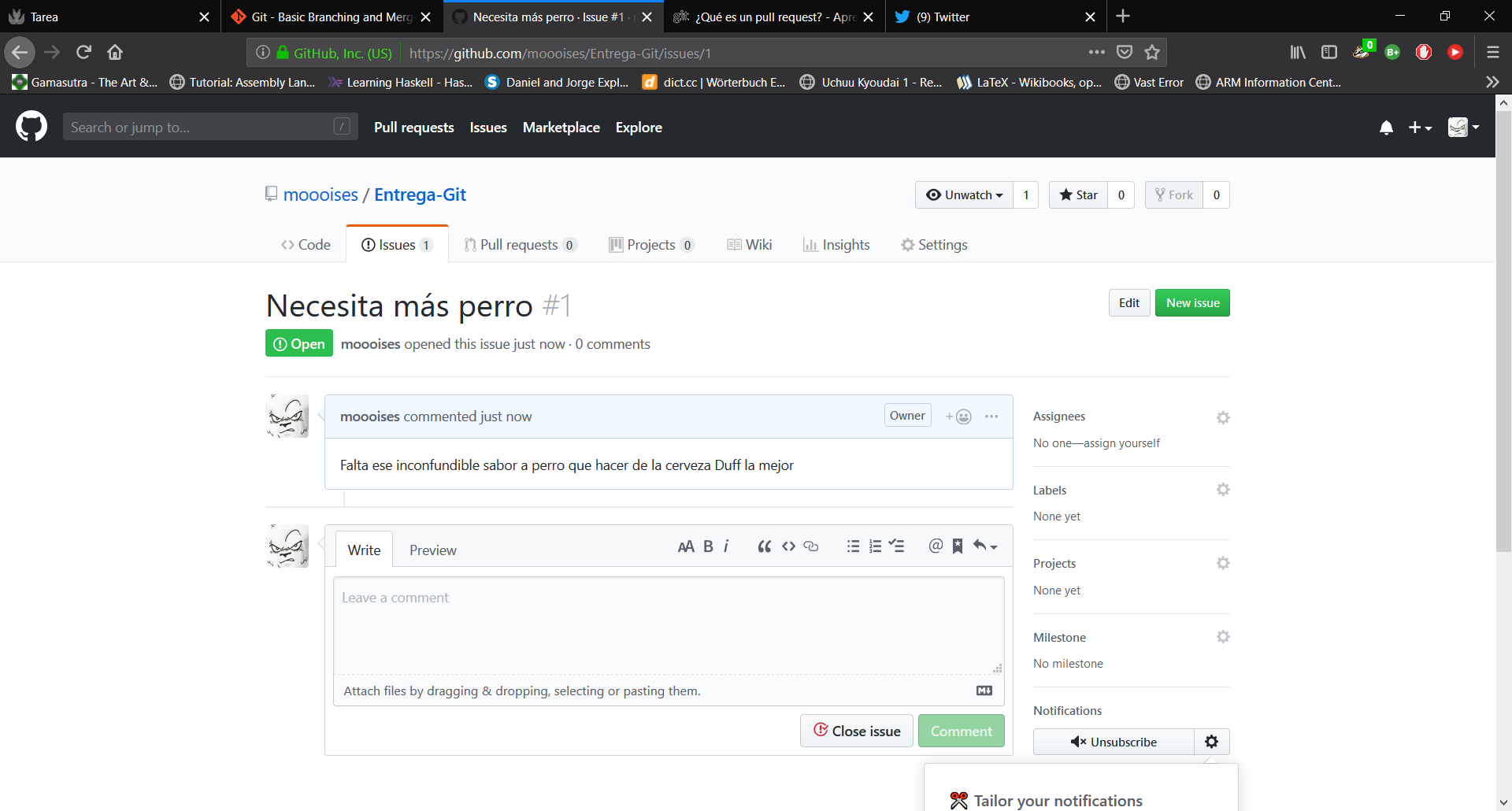
Para la versión de comandos, una vez tenemos el repositorio y los archivos nuevos o modificados, se da el comando *‘git status’* para verificarlo (este paso no es obligatorio, pero si recomendado), luego *‘git add [nombre de los ficheros]’*, seguido de *‘git commit -m “Comentario”’*, y se finaliza con *‘git push’*.

En esta imagen vemos que al realizar el “push”, nos manda un error. Y es que los compañeros estuvieron realizando cambios en el repositorio, por lo que los archivos locales que tenía no eran los más recientes. Se actualiza el repositorio local con *‘git pull’*, y se realiza el “push” para finalizar.



Usamos el comando ‘git rm’ para borrar un archivo del repositorio , sin olvidar hacer ‘git commit’ y ‘git push’.

Para tratar las issue, basta con añadirla en el repositorio de GitHub en la pestaña ‘Issue’ y a ‘New Issue’

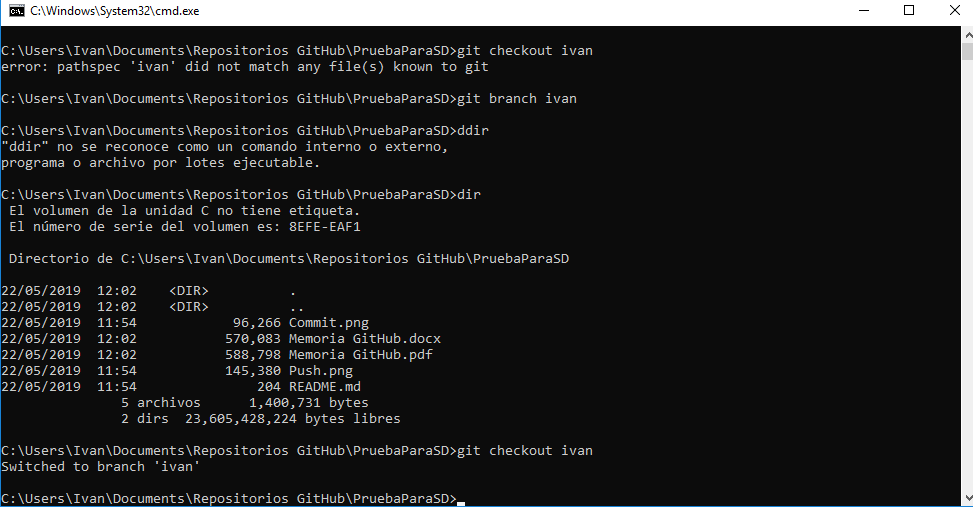


Finalmente, trataremos el tema de las ramas (en inglés ‘branch’). Las ramas nos permiten trabajar paralelamente en un mismo repositorio, sin afectar a los demás usuarios. Tenemos la rama ‘master’ que es la principal, mas todas las que queramos crear. Para crear una rama se utiliza la orden

“git branch “nombre\_rama” “

Y para cambiar de una rama a otra

“git checkout “nombre\_rama” “



Finalmente, una vez hayamos completado el trabajo que realizamos en la rama, podemos fusionarla con la rama master para incorporar en master las mejoras de la rama. Para eso nos situamos en la rama master y ejecutamos la orden:

“git merge “nombre\_rama””

Importante, una vez creada la rama es importante notificárselo a github, para eso usamos la orden

“git push --set-upstream origin nombre\_rama”

