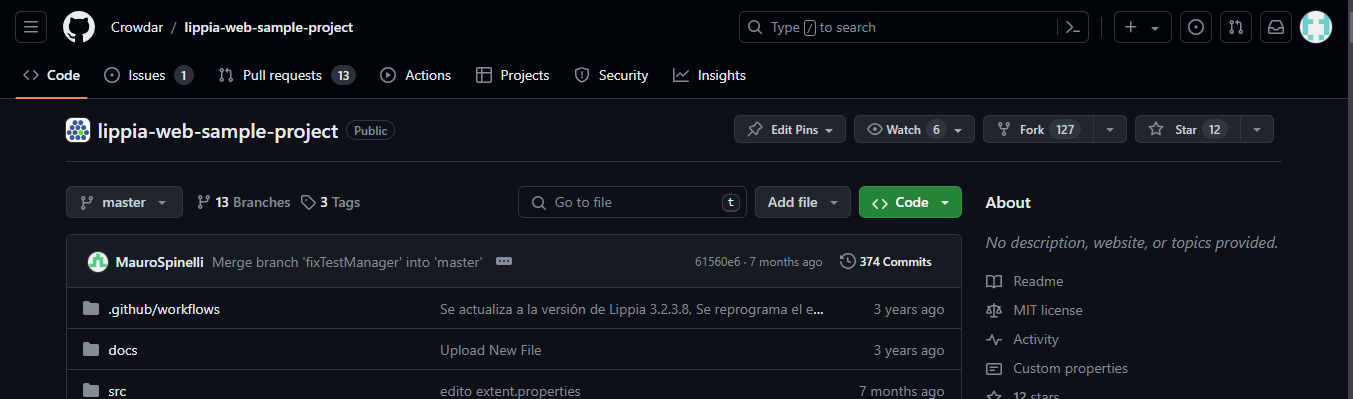
**Uso básico de Github Desktop**

1. Forkear el repo que queremos modificar.

-Estoy en la página principal donde se encuentra el archivo que quiero añadir a nuestro repositorio.

-Selecciono el botón que está arriba a la derecha que dice “Fork”.

-Ahora este repositorio esta clonado en nuestra cuenta y podemos hacer uso libre de él.

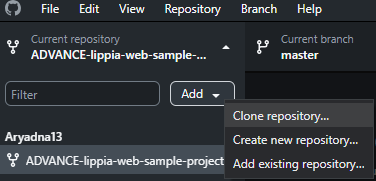


2-Clonar el repo en nuestra PC

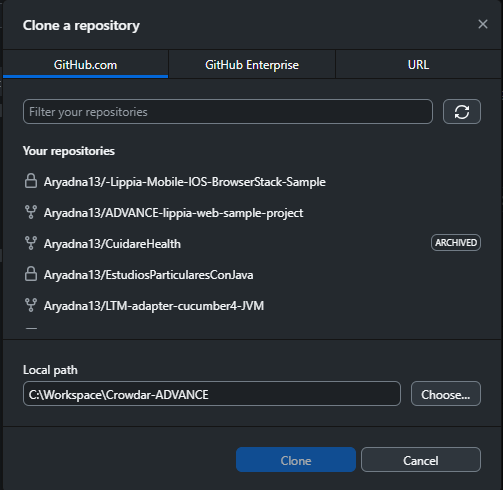
-Estoy sobre la página principal y en la rama máster de MI repositorio en MI CUENTA para chekear que ha sido forkeado correctamente el repositorio en el que deseo trabajar.

-Abro GitHub Desktop para continuar.

-Clickeo el botón “Add” y seguido la opción “Clone repository”.

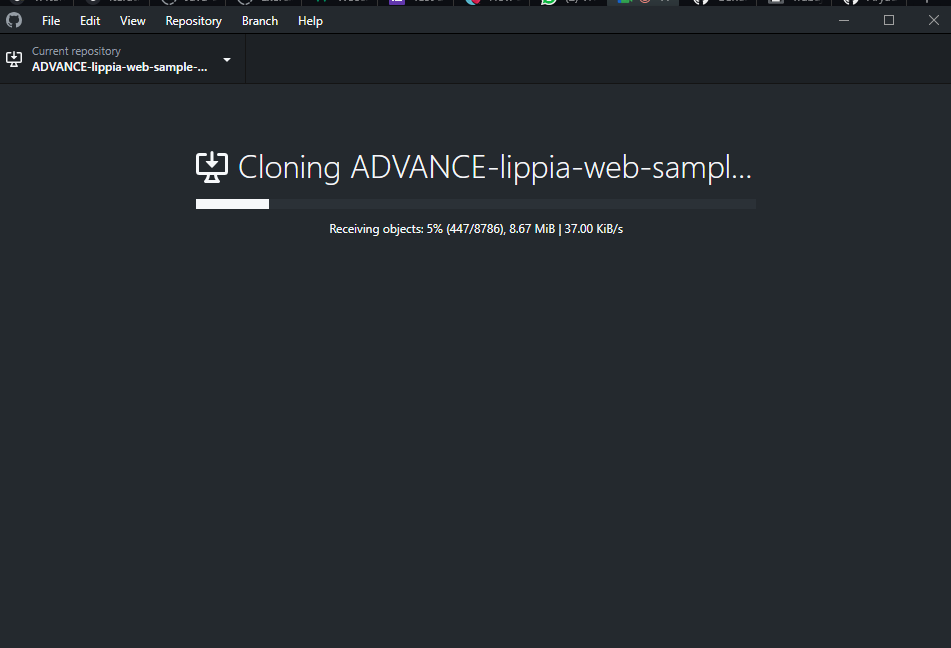


-Se despliega un menú con todos los repositorios existentes en nuestra cuenta. Aquí selecciono el que quiero clonar para poder trabajar en mi PC.



-Lo selecciono, marco la ruta donde lo quiero dejar guardado y le doy al botón “Clone”.

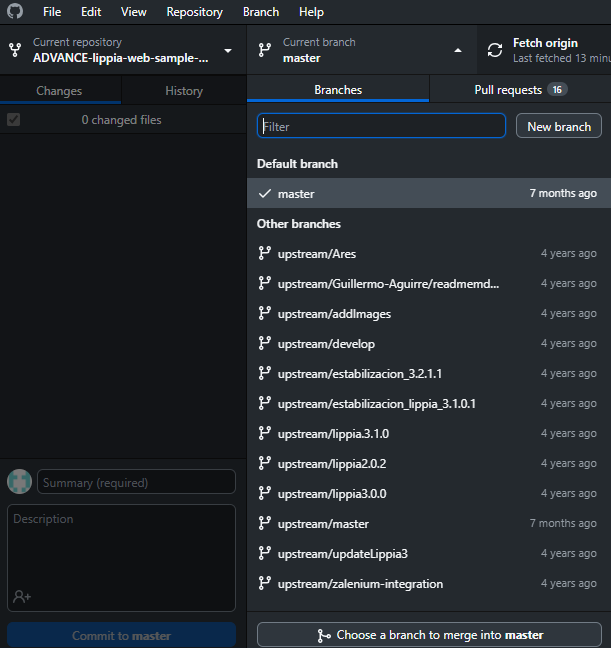
-Se iniciará la descarga de ese repo y podre corroborar que se encuentra ahí si nos dirigimos a la carpeta donde lo guardamos.



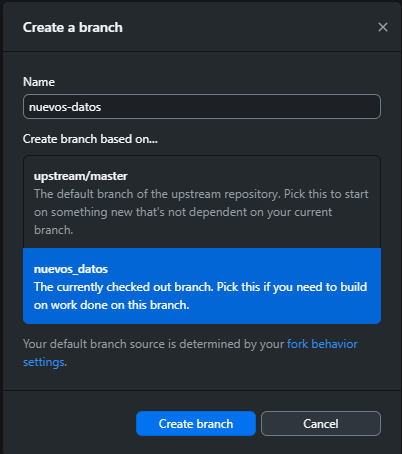
3- Creando una nueva rama “New Branch”

-Creo una nueva rama para hacer todas las modificaciones que quiera a mi gusto sin que la rama principal “Máster” sufra ningún cambio. Esto se hace por si tenemos algún conflicto en nuestro espacio de trabajo o tocamos algo que no debemos y queremos volver al principio absoluto. También para reutilizarlo si queremos experimentar.

-En Github Desktop selecciono la siguiente opción y elijo crear la nueva rama con “New Branch”.



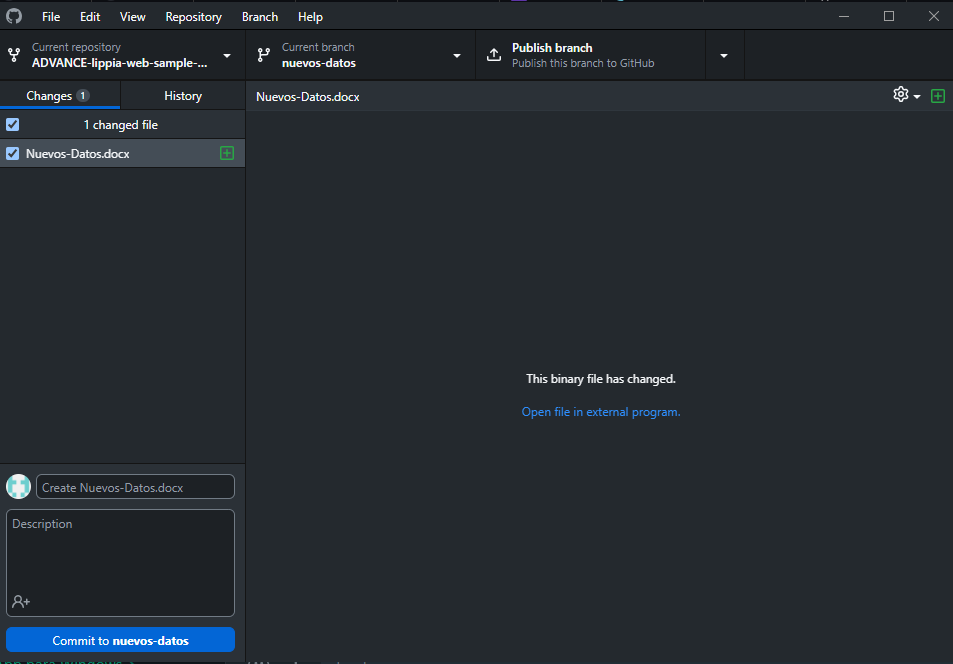
-La nombro (en este caso lo nombre “nuevos-datos”) y clickeo “Create Branch”, ahora puedo trabajar sin miedo a que pase nada sobre esa rama.



4- Agregar un archivo nuevo al repositorio

-Hago todas las modificaciones que quiero en mi espacio de trabajo creado en el pc.

-En GitHub Desktop a medida que actualice mi archivo me va a indicar los cambios hechos, es importante poner un “Commit” claro y preciso, sin puntos suspensivos o finales describiendo los cambios nuevos.

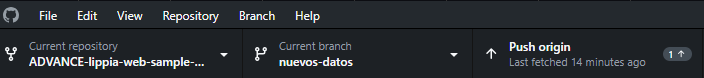


-Una vez hecha la descripción le damos al botón “Commit to”

5- Pushear los cambios hechos.

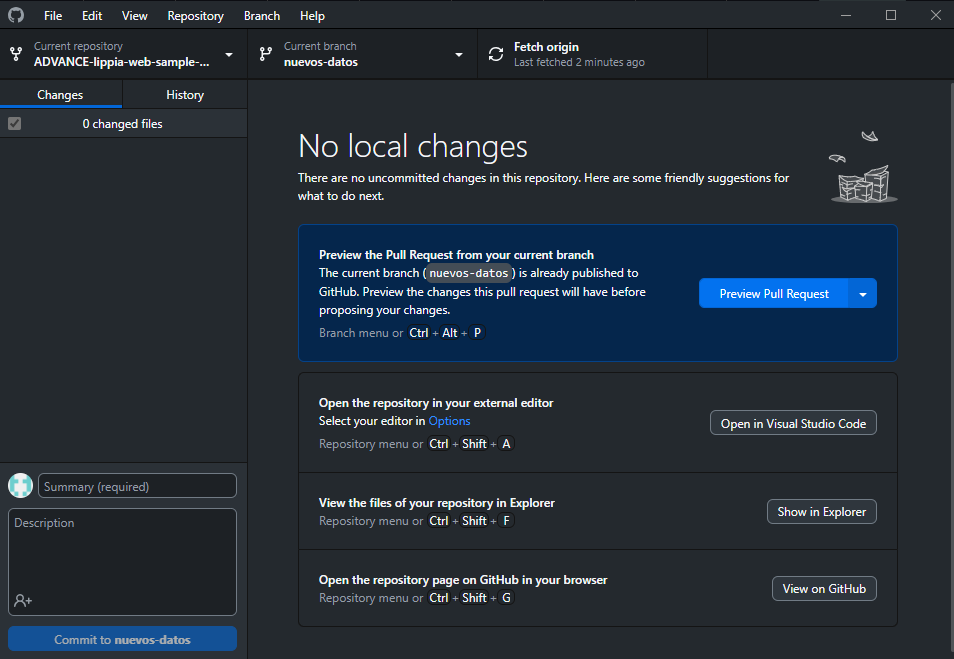
-Una vez que haya terminado todos mis cambios, lo voy a subir a mi GitHub para tenerlo disponible cuando quiera usarlo. Una vez commiteado, tengo que “pushearlo” a mi repositorio en la nube.

-Le doy al botón “Push origin” para que se suba.

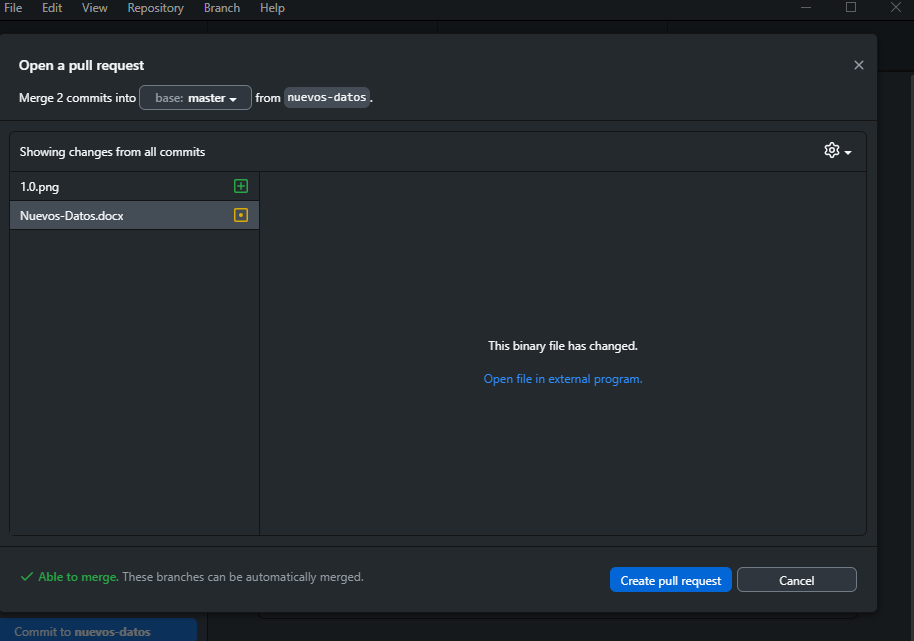


6- Pull Request

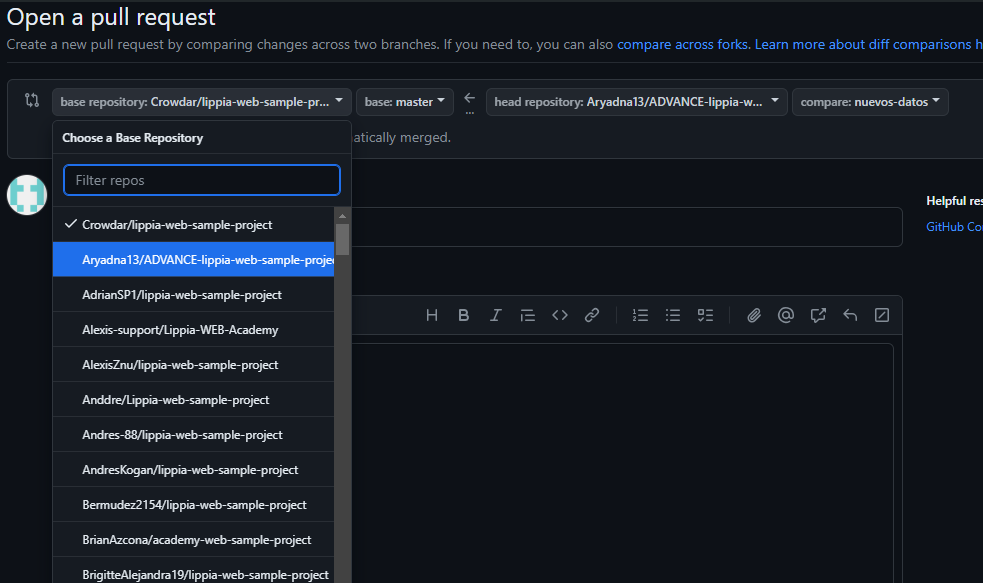
-Para forzar el cambio selecciono la opción “Preview Pull Request”



-Creo mi “Pull Request”

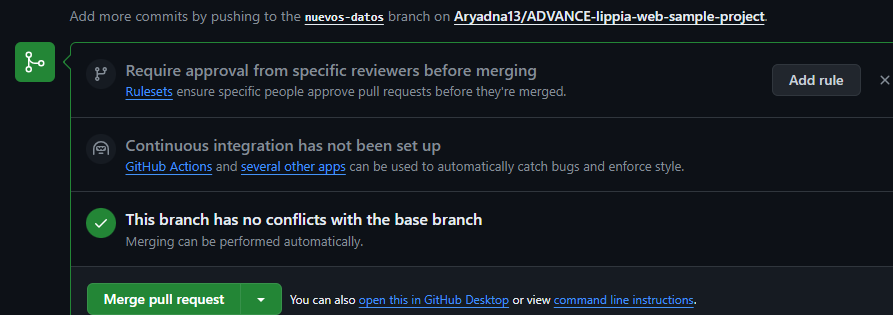


IMPORTANTE: Verificar que siempre se haga sobre la rama en la que estamos trabajando para evitar subir cambios a la rama Máster, o a cualquier otro lado.



7-Merge

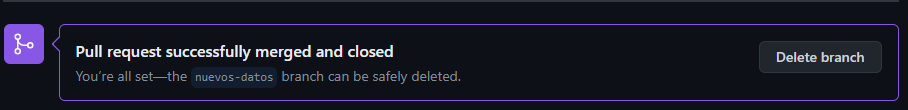
-Termino de unificar los cambios que subi con los archivos que tengo en la nube con un Merge.



Captura de pantalla de un celular

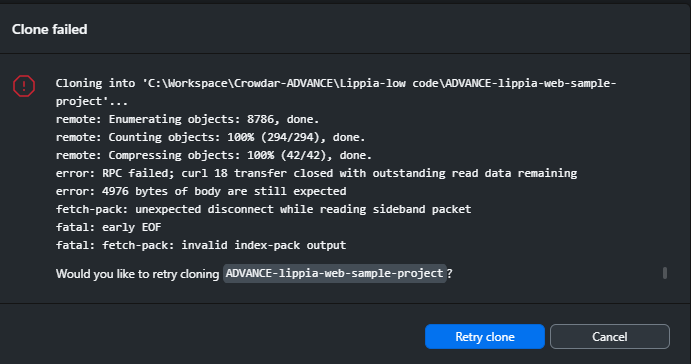
Descripción generada automáticamente

-Asi se ve el merge exitoso.



Extras:

A veces una mala conexión de internet puede provocar una pérdida de paquetes a la hora de clonar un repo, este cartel de error es prueba de ello. Si bien lo probé hacer desde la misma terminal de Git Bash me salía el mismo cartel de error.



Simplemente lo descargué en otro momento del día en que mi conexión era más estable.