TRABAJO DE INVESTIGACIÓN

SOBRE GIT DE ESCRITORIO



Entornos de desarrollo

David Guijo López 17/03/2023

ÍNDICE

1º- Creación de una cuenta de GitHub

2º-Instalación de de GitHub de escritorio

3º-Creación de un repositorio

4º-Publicar repositorio

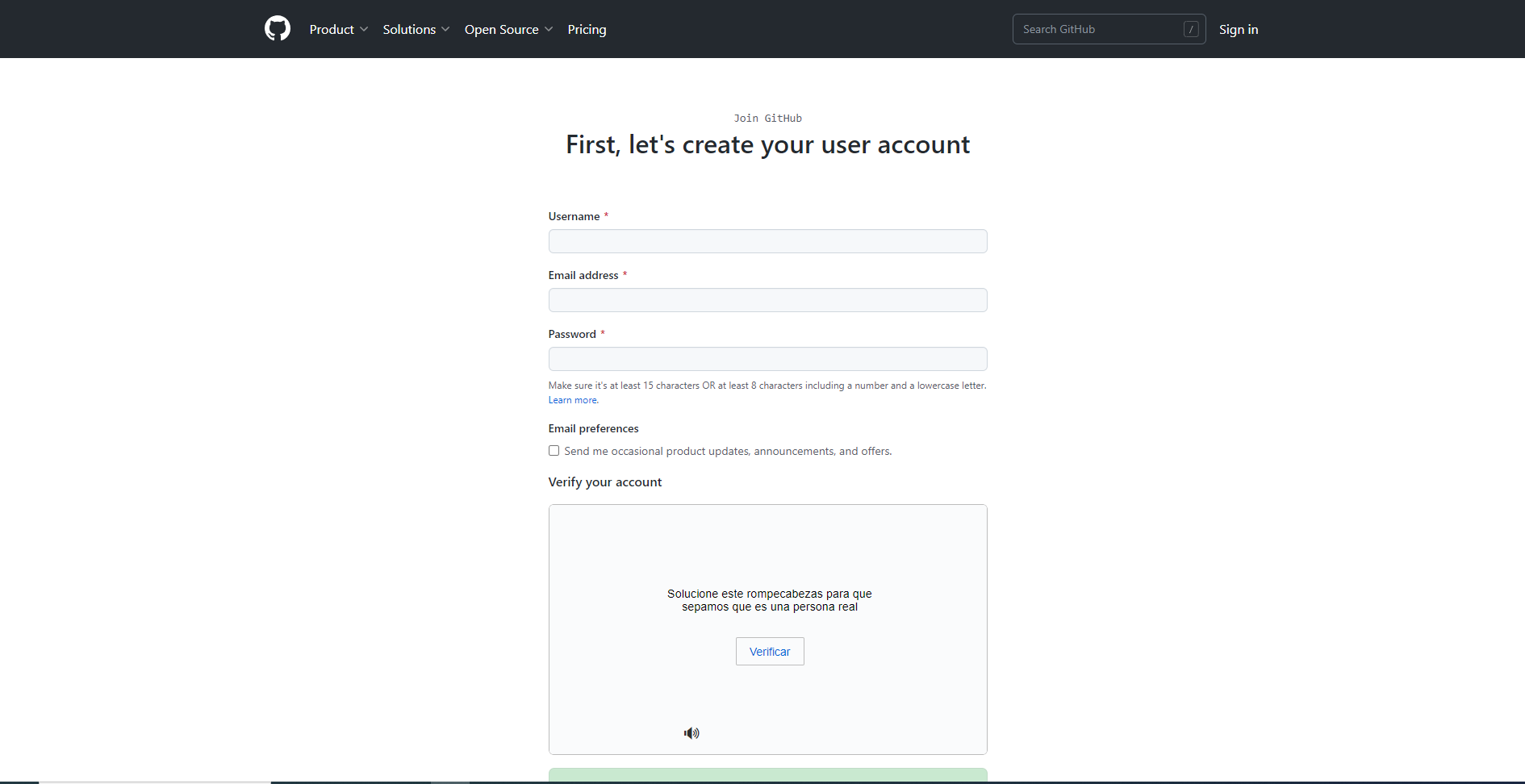
5º-Trabajar con un repositorio

**1- CREACIÓN DE LA CUENTA DE GITHUB**

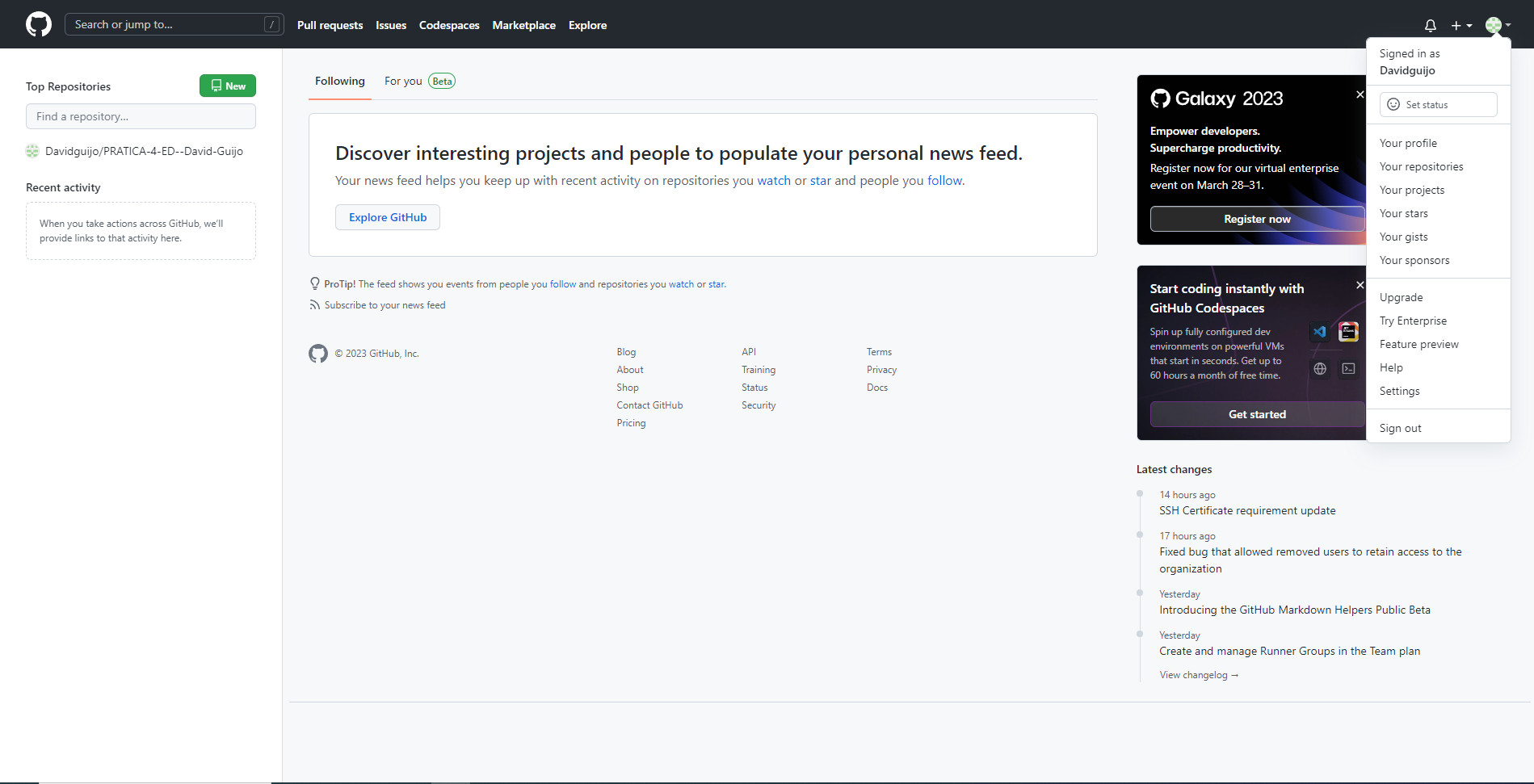
Para crear nuestra cuenta de GitHub debes seguir los siguientes pasos:

1-En primer lugar debes clicar en el enlace a continuación y rellenar los siguientes datos para crear tu cuenta

<https://github.com/join>



2- Tras iniciar sesión podremos ver que nuestro perfil está situado en la parte derecha



Ya tenemos nuestra cuenta de GitHub creada

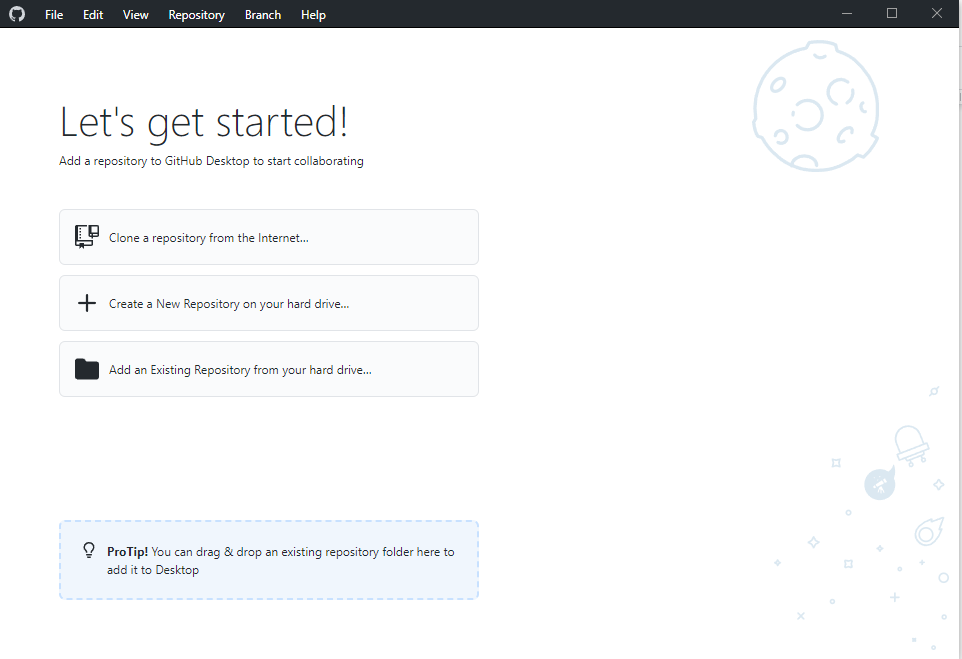
**2- INSTALACIÓN DE GITHUB DE ESCRITORIO**

Para instalar GitHub de escritorio debemos de seguir los siguientes pasos:

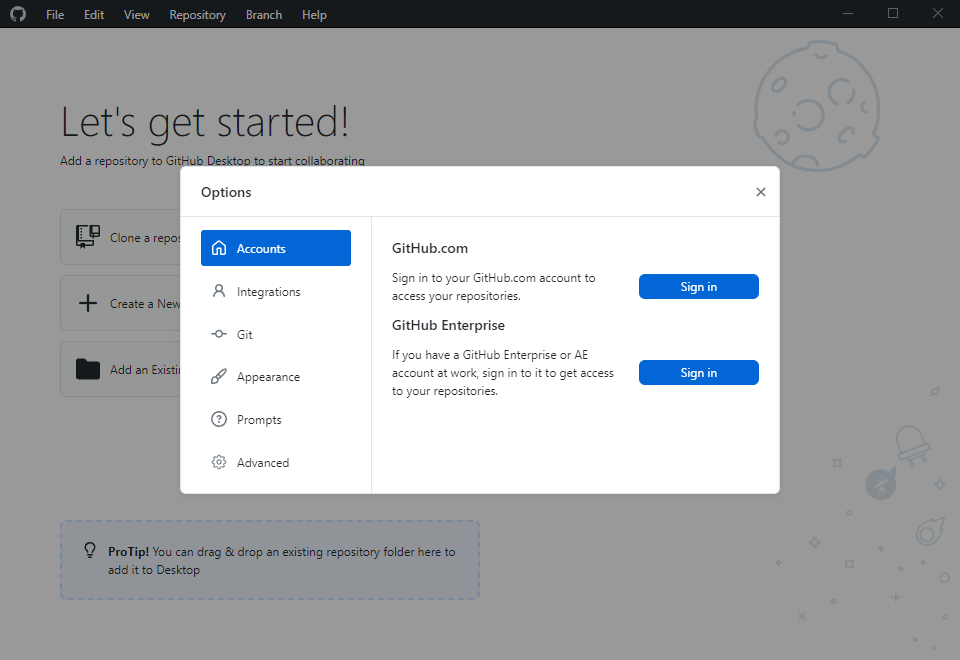
1-Clicamos en el enlace a continuación y le damos a descargar para Windows

<https://desktop.github.com/>

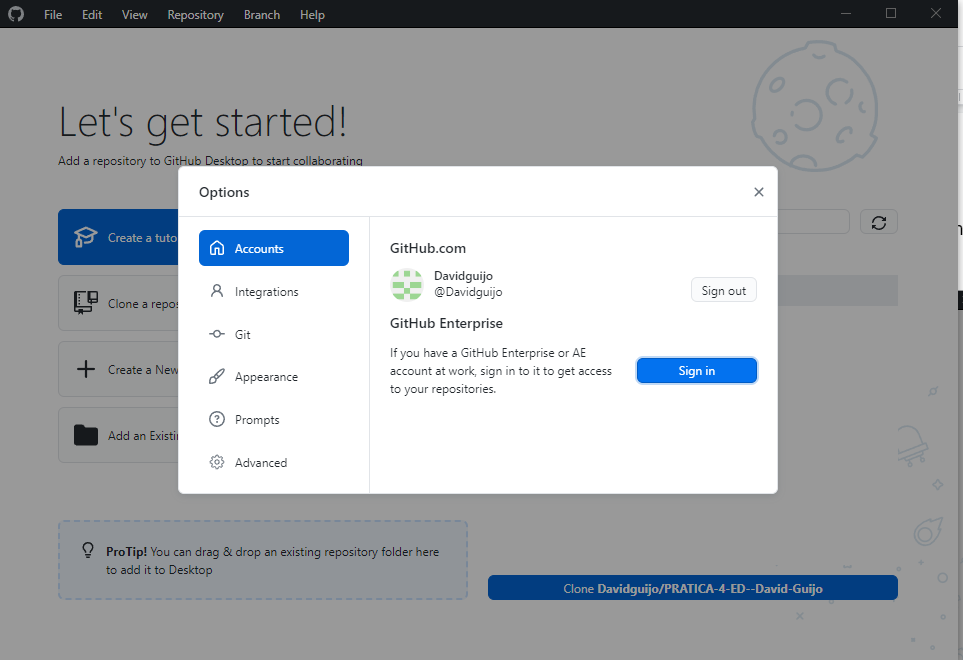
2-Cuando se haya descargado GitHub abrimos la aplicación haciendo doble clic sobre ella, lo que debemos de ver es algo así



3-Clicamos en la pestaña File y le damos a Options, con el objetivo de crearnos una cuenta de GitHub



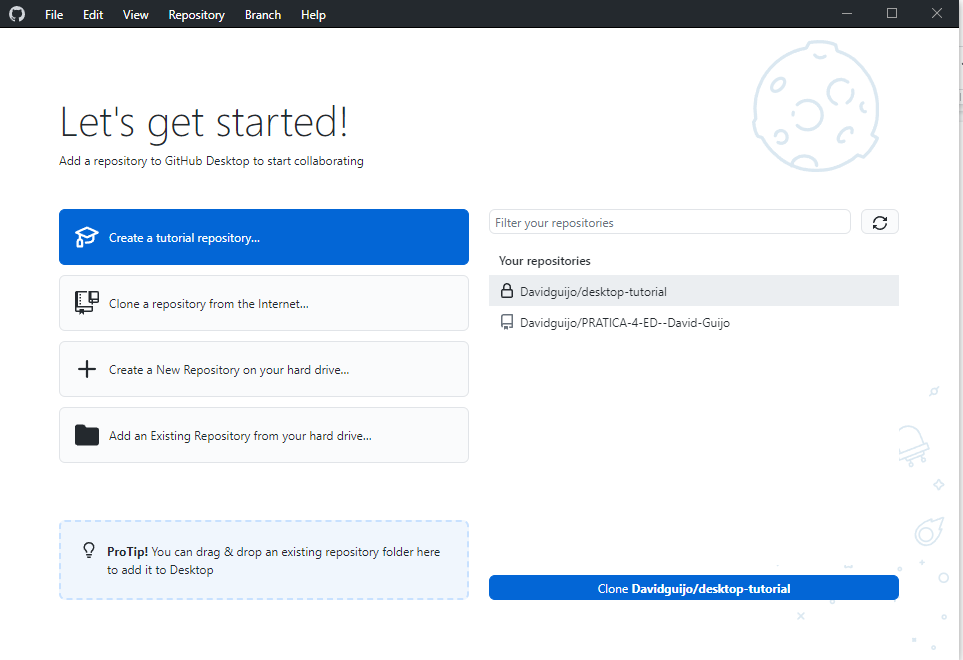
4-Para registrarnos clicamos en Sign in, dicho enlace nos llevará a una web con la que GitHub enlazará nuestro perfil con la aplicación



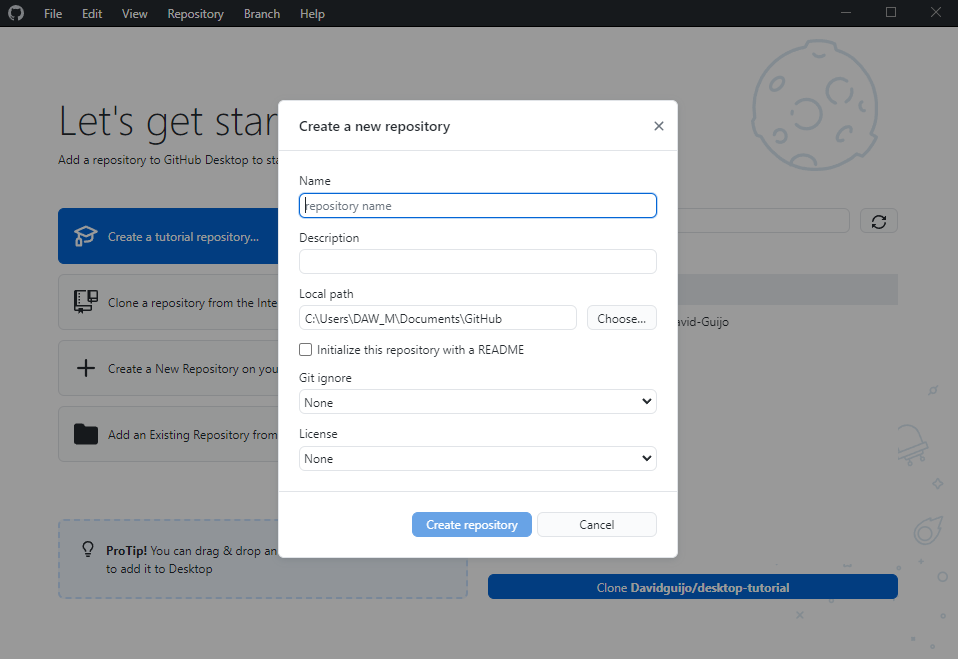
Como puedes comprobar, mi perfil ya está enlazado con mi cuenta de GitHub

**3- CREACIÓN DE REPOSITORIO**

Tenemos múltiples opciones para crear un repositorio, podemos crear o clonar repositorios de tutoriales, clonar un repositorio de internet, crear un nuevo repositorio o añadir un repositorio existente desde el disco duro

****

1-Para crear un repositorio clicamos en Create a New Repository on your hard drive y rellenamos todos los datos



-Name: Nombre del repositorio

-Description: Es un campo opcional con el que puedes brindar mayor información de tu repositorio

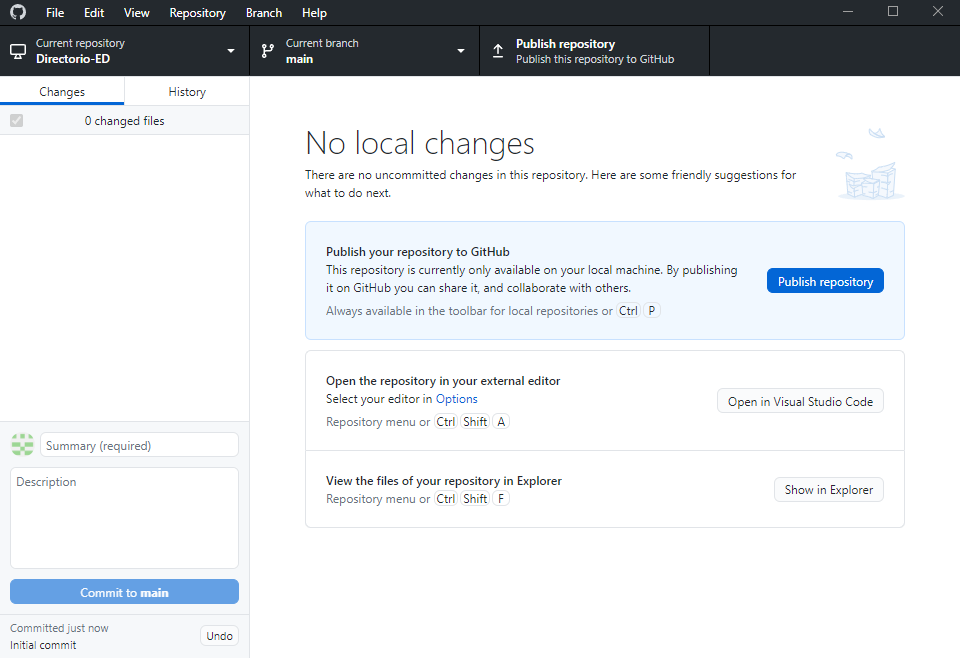
-Local path: Establece la ubicación del repositorio en tu ordenador

-Initialize this repository with a README: crea una confirmación inicial con un archivo README, que ayudará a las personas a comprender el objetivo de tu proyecto

-Git ignore: permite agregar un archivo personalizado para ignorar archivos concretos del repositorio local que no quiera almacenar en el control de versiones

-License: permite agregar una licencia de código abierto a un archivo LICENSE en el repositorio

2-Una vez creado nuestro repositorio, en la pantalla se nos mostrará algo así



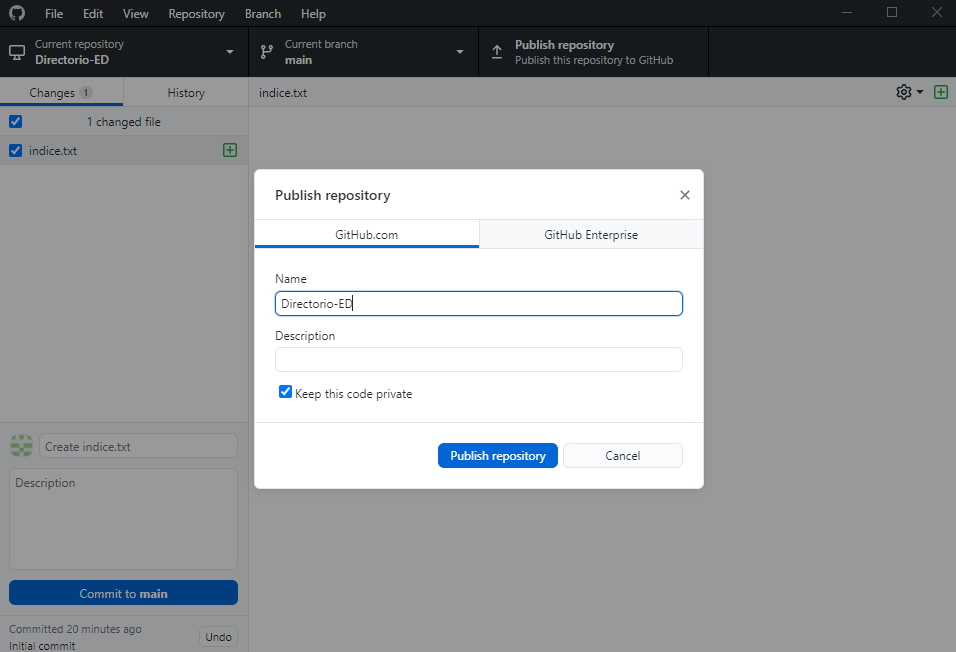
En la parte superior se nos indica el nombre del repositorio actual, la rama en la que nos encontramos y una opción para publicar nuestro repositorio.

En la parte de la izquierda nos aparecen las opciones cambios e historial, en cambios se van a mostrar los cambios realizados pero que aún no se han confirmado, y en historial muestra las confirmaciones que hayamos realizado

**4- PUBLICAR REPOSITORIO**

Para publicar nuestro repositorio debemos de seguir los siguientes pasos:

1-Clicamos en la pestaña de publicar repositorio



El nombre y la descripción se rellenarán automáticamente (en mi caso no tengo ninguna descripción del repositorio)

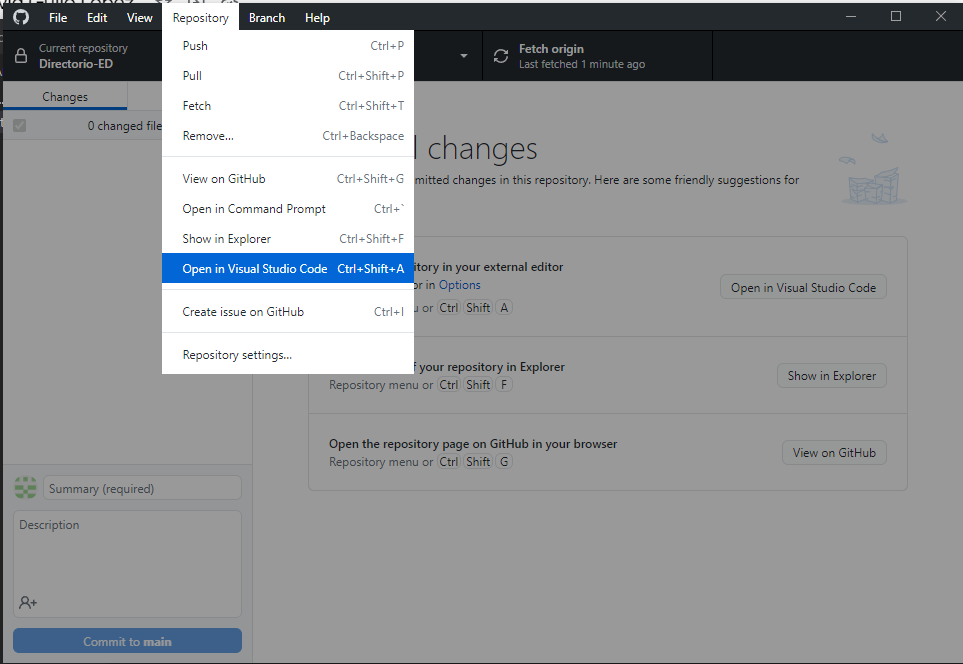
También vemos que podemos marcar la opción Keep this code private, si no la seleccionas cualquier usuario de Git podrá ver su contenido, en cambio si seleccionas esta opción tu código estará disponible solo para tí

En la pestaña GitHub enterprise podremos publicar el repositorio en una organización concentra de GitHub a la que se pertenezca

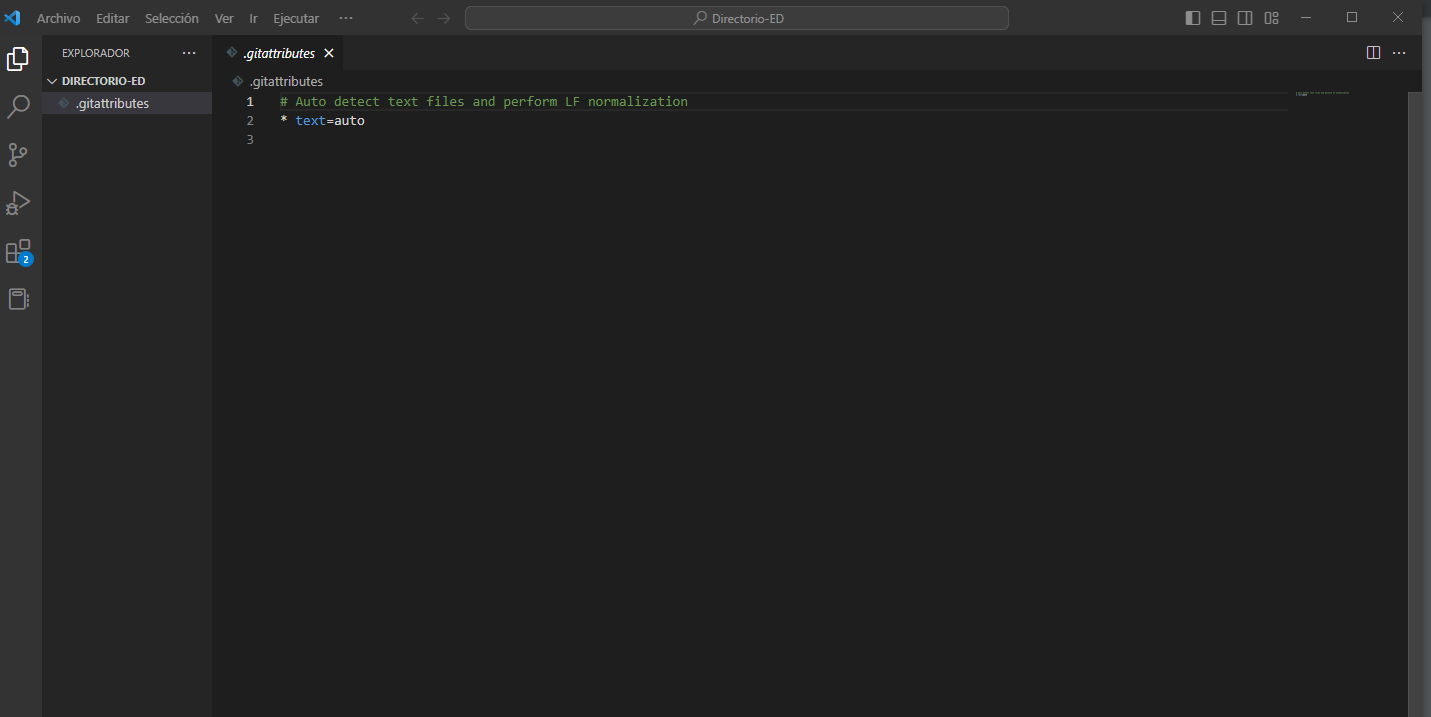
2-Una vez visto todo esto, para publicar el repositorio clicamos en Publish repository

**5- TRABAJAR CON UN REPOSITORIO**

1 -Para empezar a trabajar con un repositorio, clicamos en la pestaña Repository y en la opción Open in Visual Studio Code, esta acción nos abrirá el programa Visual Studio Code en nuestro ordenador



Ahora mismo mi archivo de Visual Studio esta vació

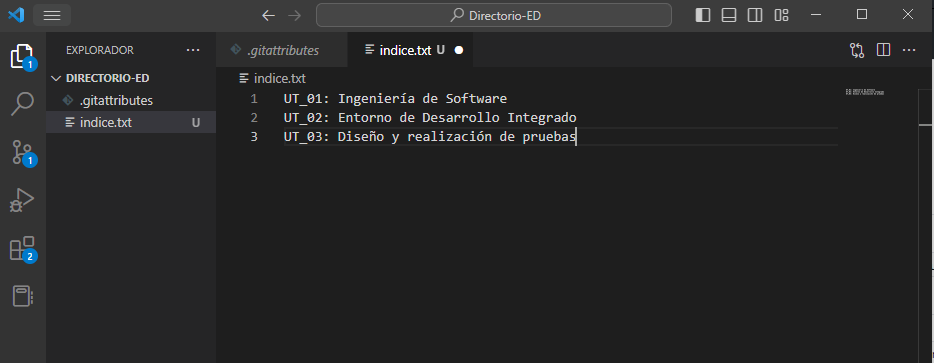


1-Creamos el fichero indice.txt e introducimos el siguiente contenido

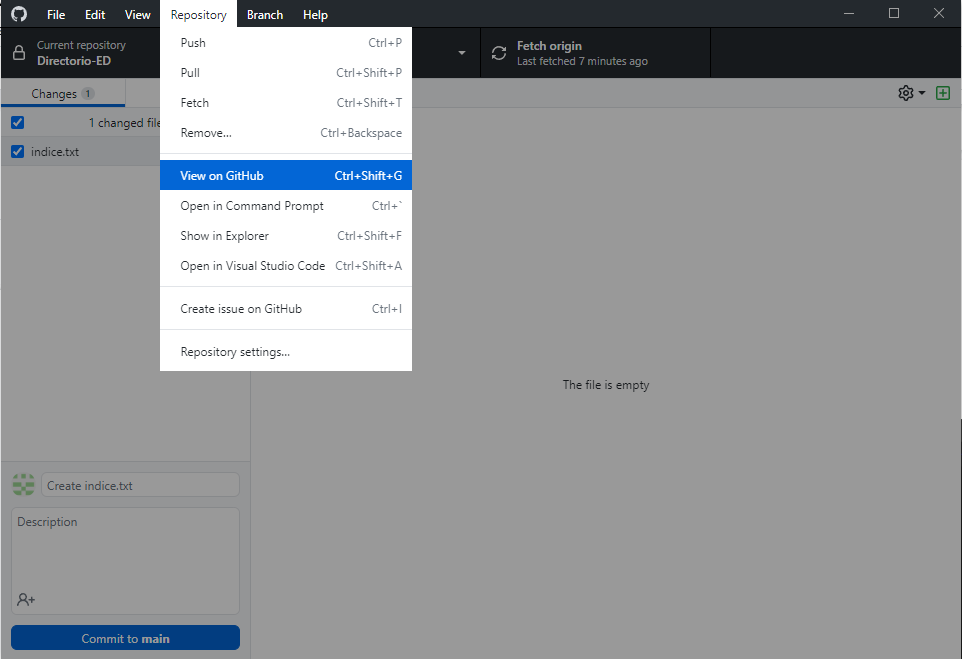
UT\_01: Ingeniería de Software

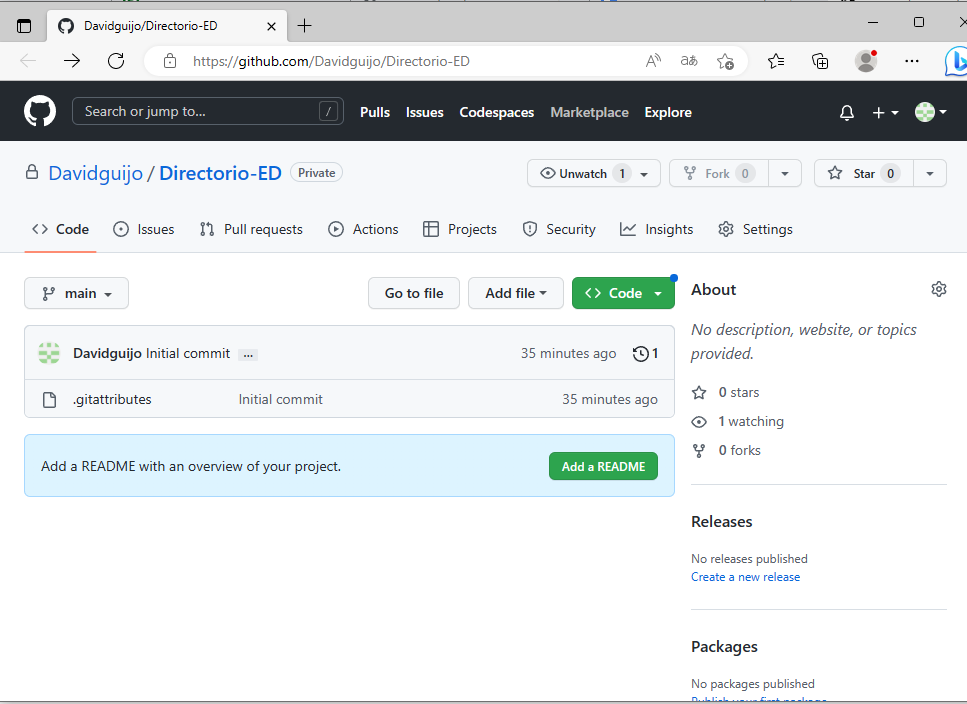
UT\_02: Entorno de Desarrollo Integrado

UT\_03: Diseño y realización de pruebas

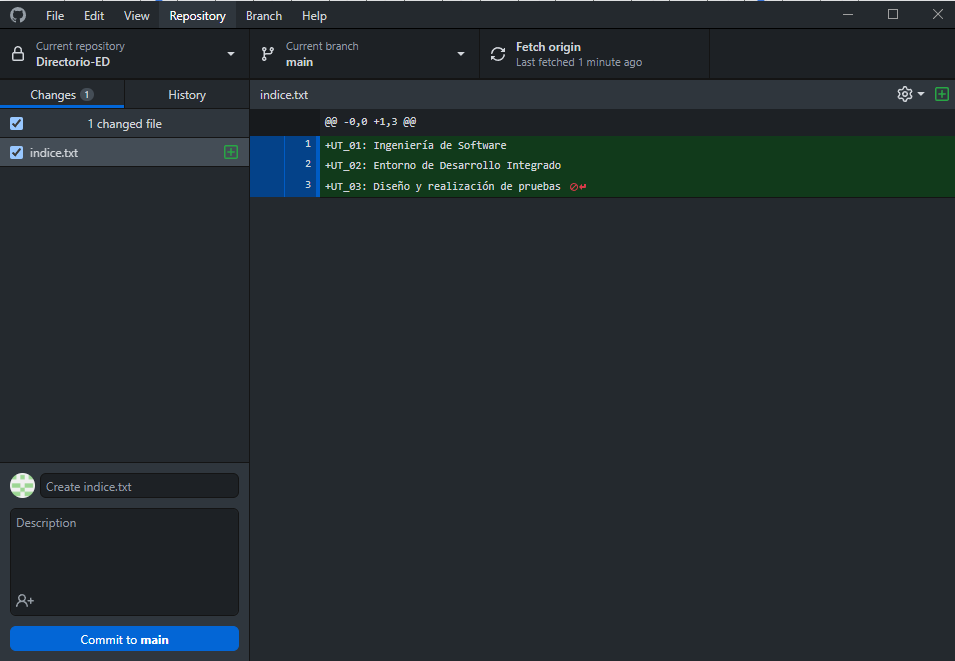


2-Para ver el estado de repositorio volveremos a situarnos en el GitHub Desktop, clicamos en la pestaña Repository y le daremos donde dice View on GitHub





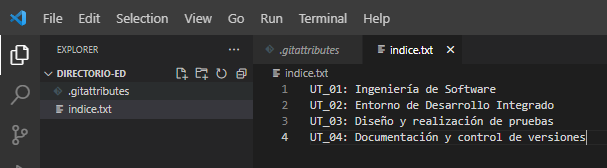
Este es el directorio que hemos creado, en el símbolo del reloj (el que esta a la derecha de 35 minutes ago) se nos indica una lista de las confirmaciones del repositorio GitHub.

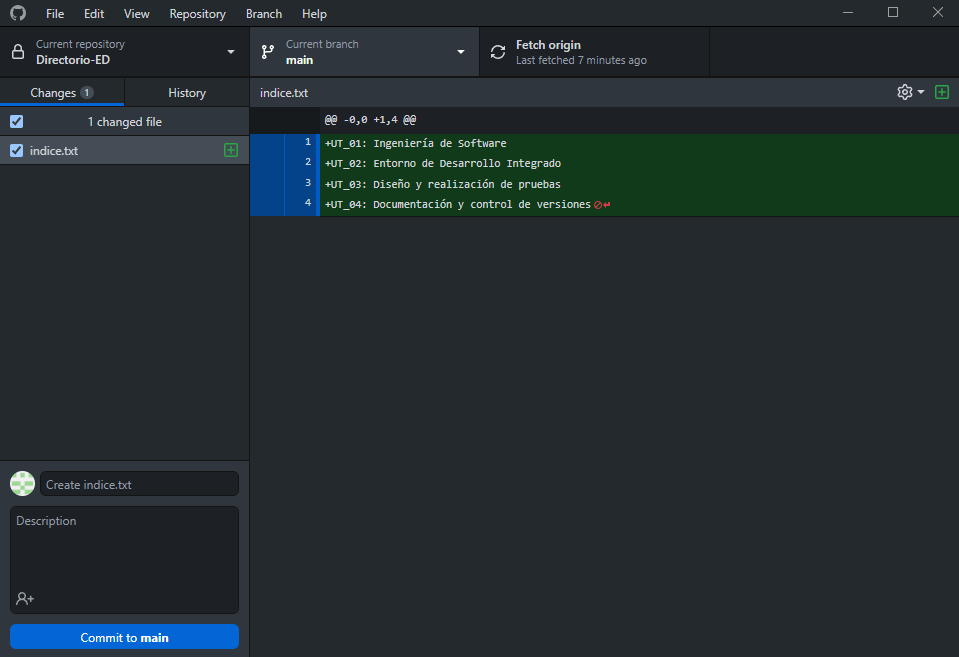


Hemos guardado los cambios y podemos verlo en GitHub

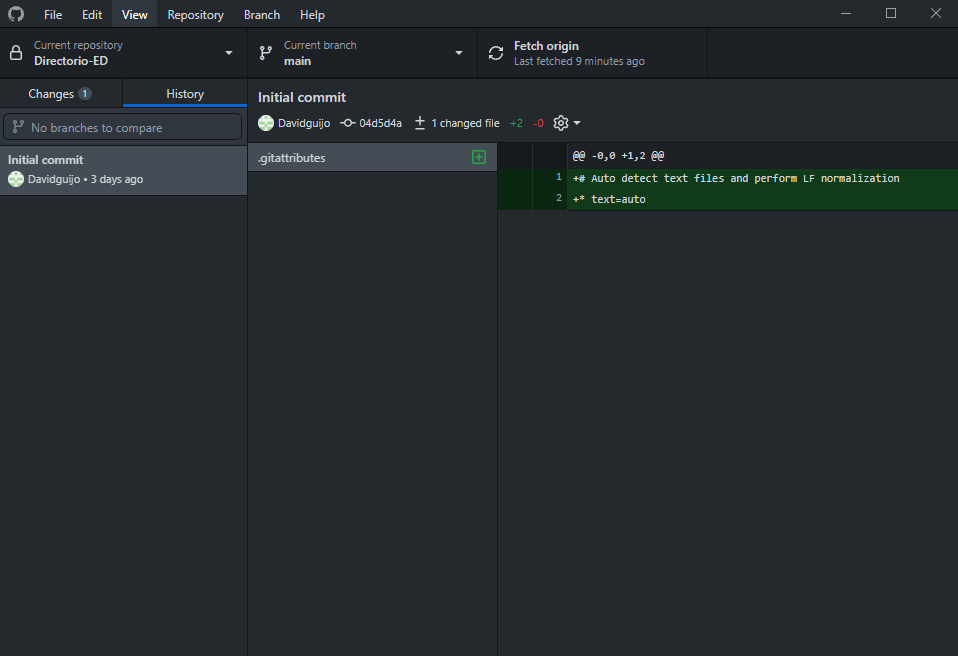
Añado lo siguiente: “UT\_04: Documentación y control de versiones”

y vuelvo a comprobar el estado



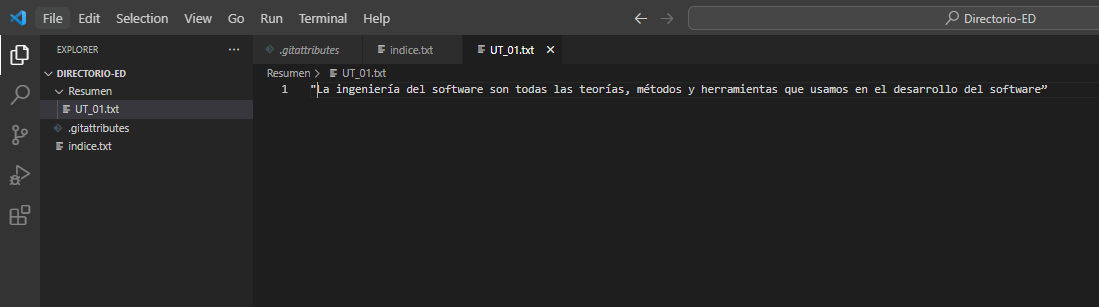


Muestro el historial de los cambios realizados

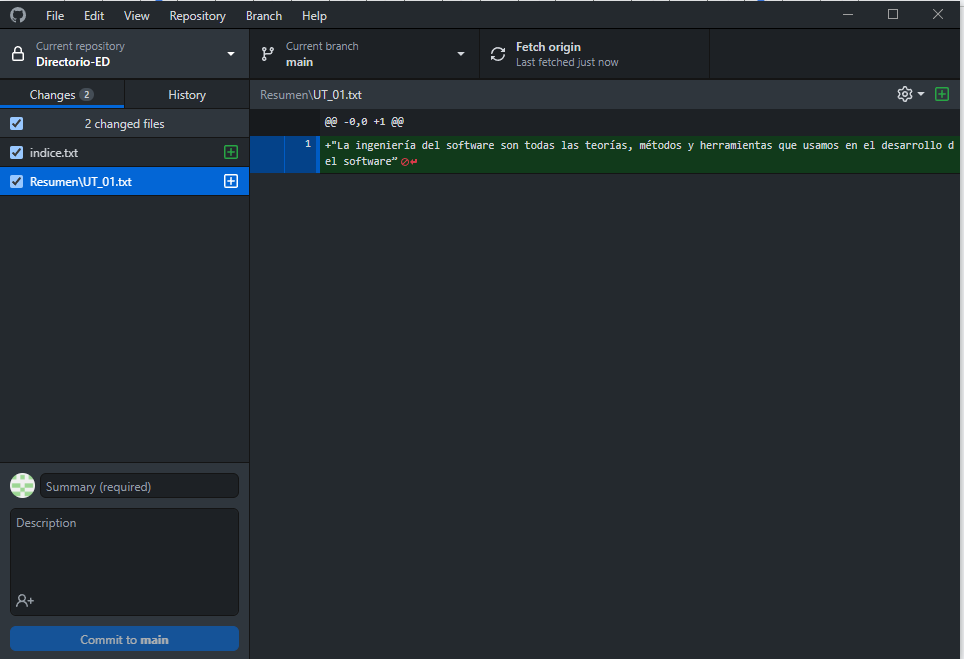


Creo una carpeta llamada Resumen y dentro de ella el fichero UT\_01.txt con el contenido :

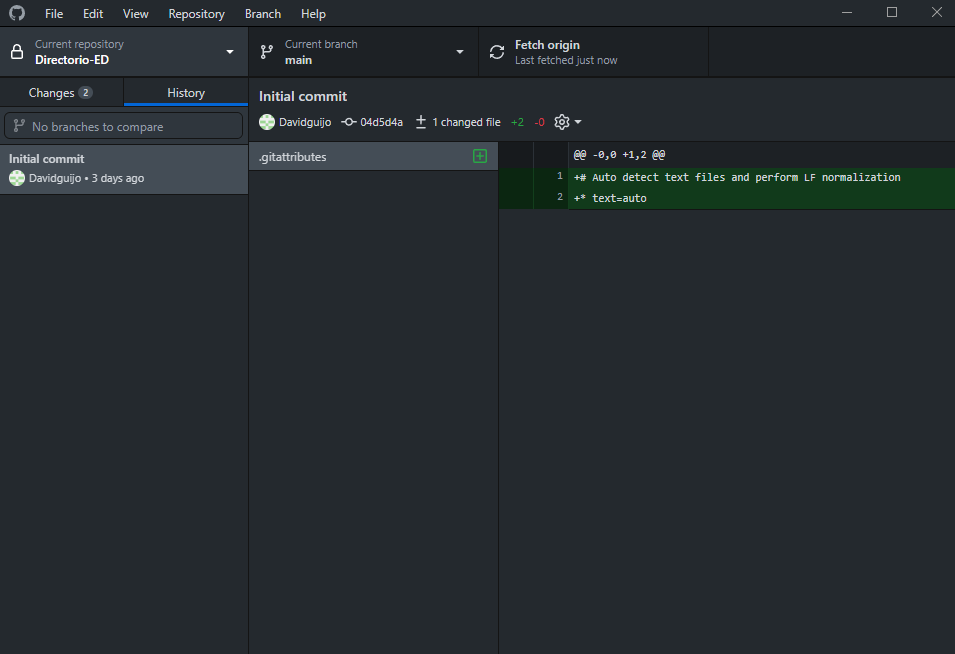
“La ingeniería del software son todas las teorías, métodos y herramientas que usamos en el desarrollo del software”



Compruebo los cambios

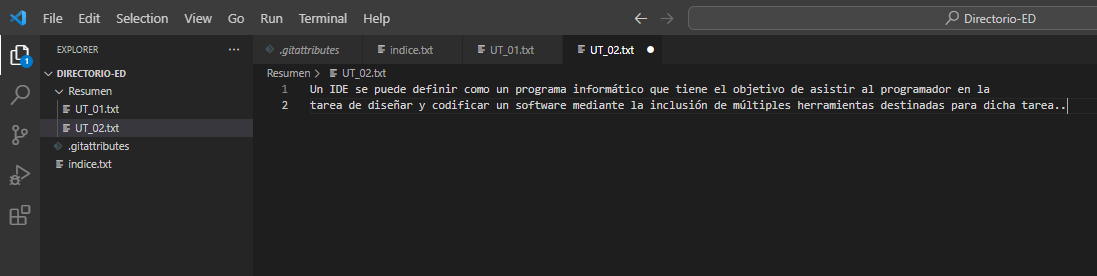


Compruebo el historial de cambios

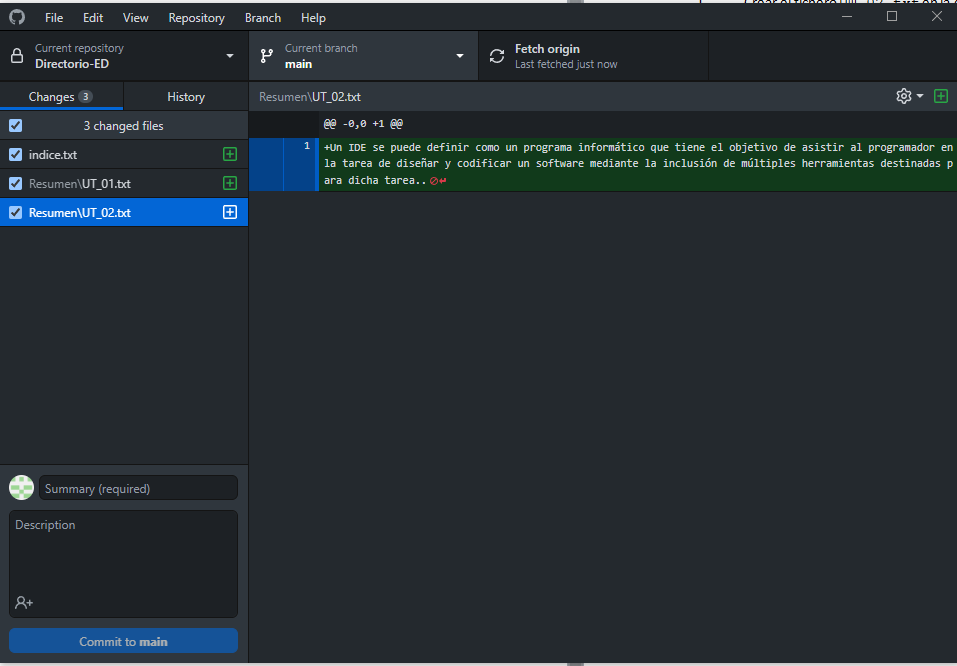


Creo un fichero UT\_02.txt con el contenido:

“Un IDE se puede definir como un programa informático que tiene el objetivo de asistir al programador en la tarea de diseñar y codificar un software mediante la inclusión de múltiples herramientas destinadas para dicha tarea..”

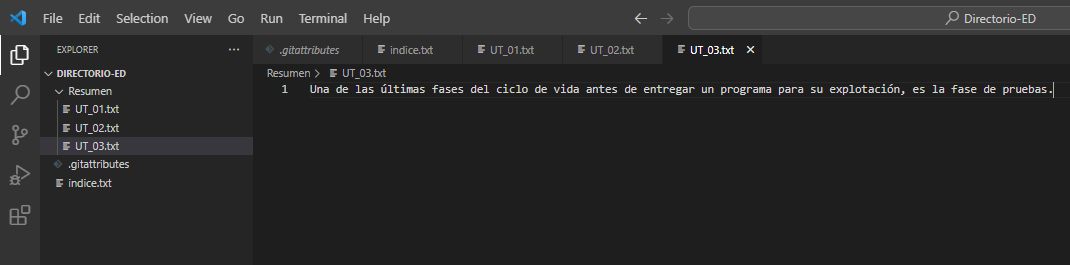


Muestro los cambios:

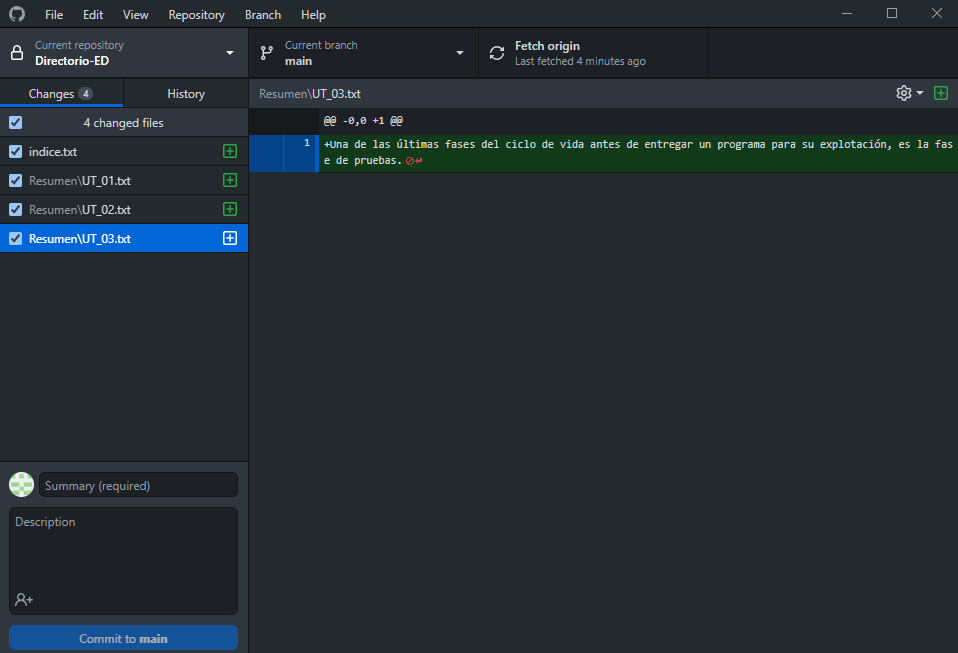


Creo un fichero UT\_03.txt con el contenido:

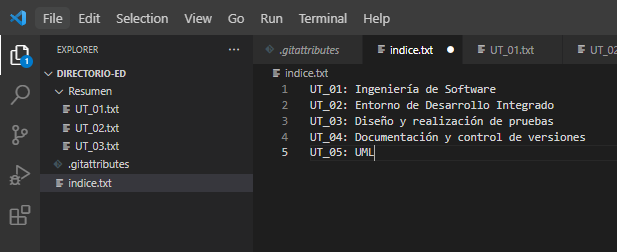
“U*na de las últimas fases del ciclo de vida antes de entregar un programa para su explotación, es la fase de pruebas*.”



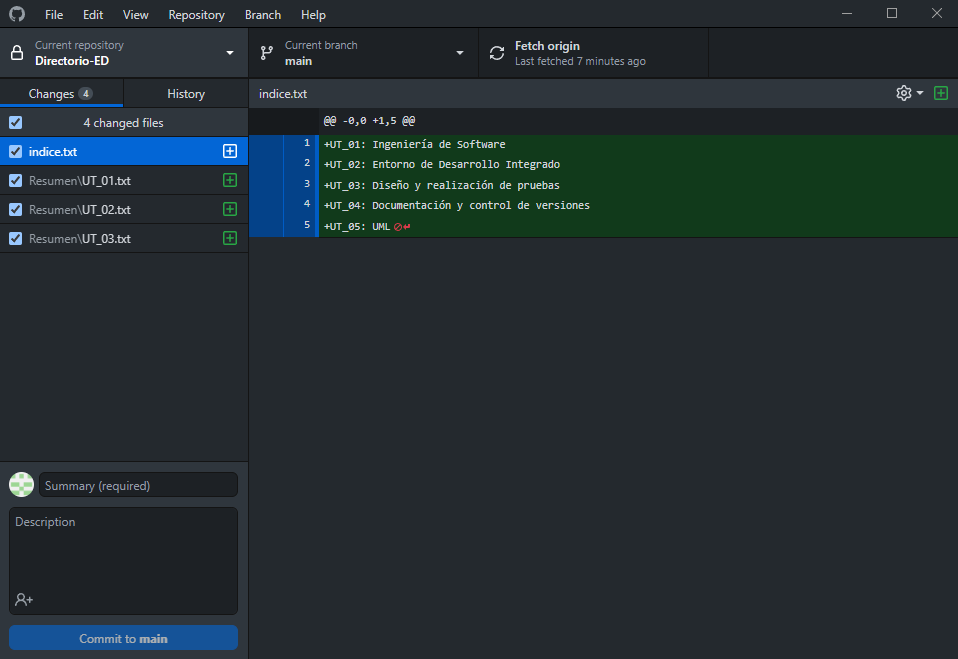
Compruebo los cambios:



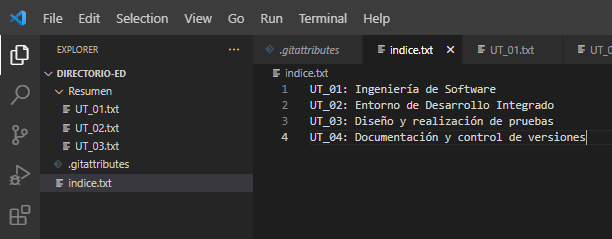
Añado al fichero indice.txt la siguiente linea: “UT\_05: UML”



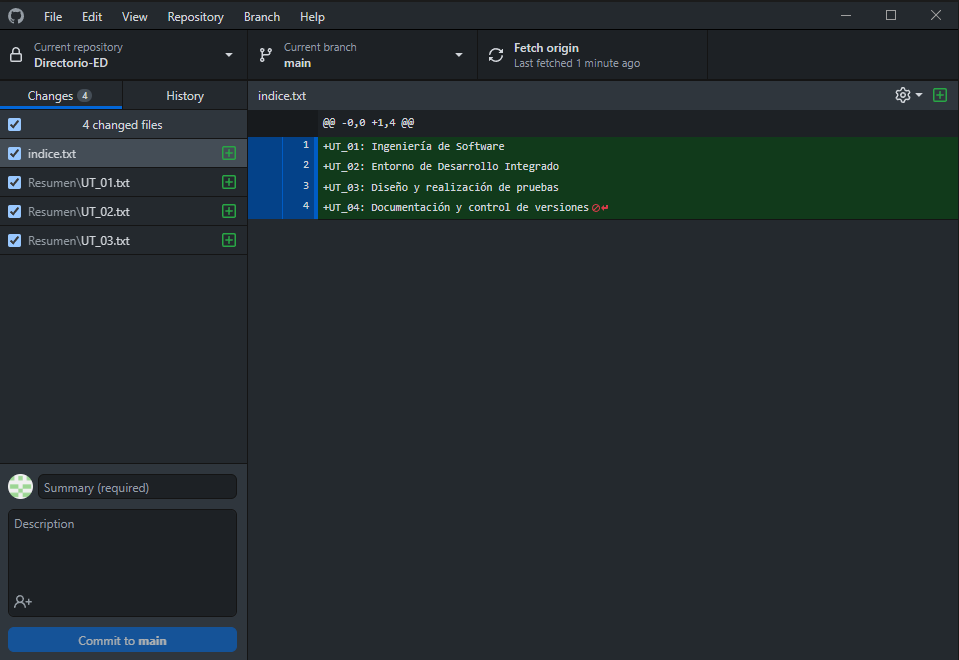
Muestro los cambios de fichero



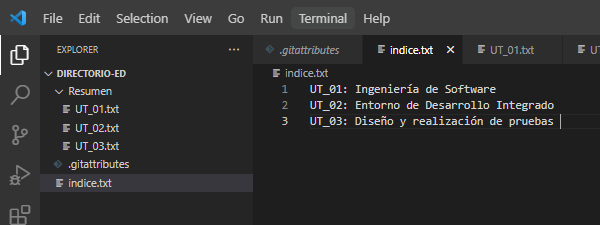
Elimino la ultima línea del fichero indice.txt



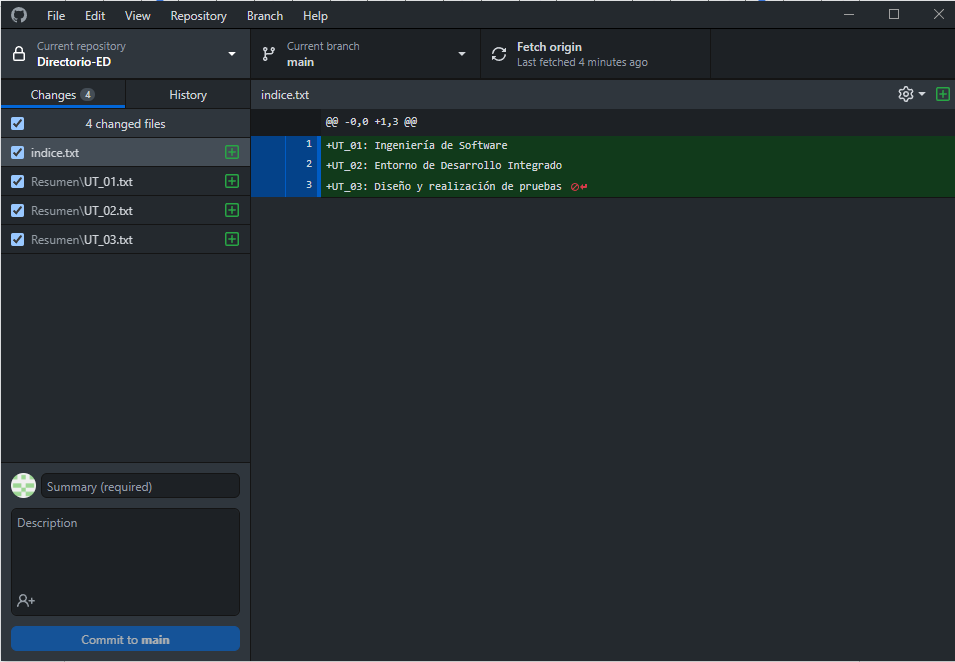
Vuelvo a mostrar los cambios del repositorio:



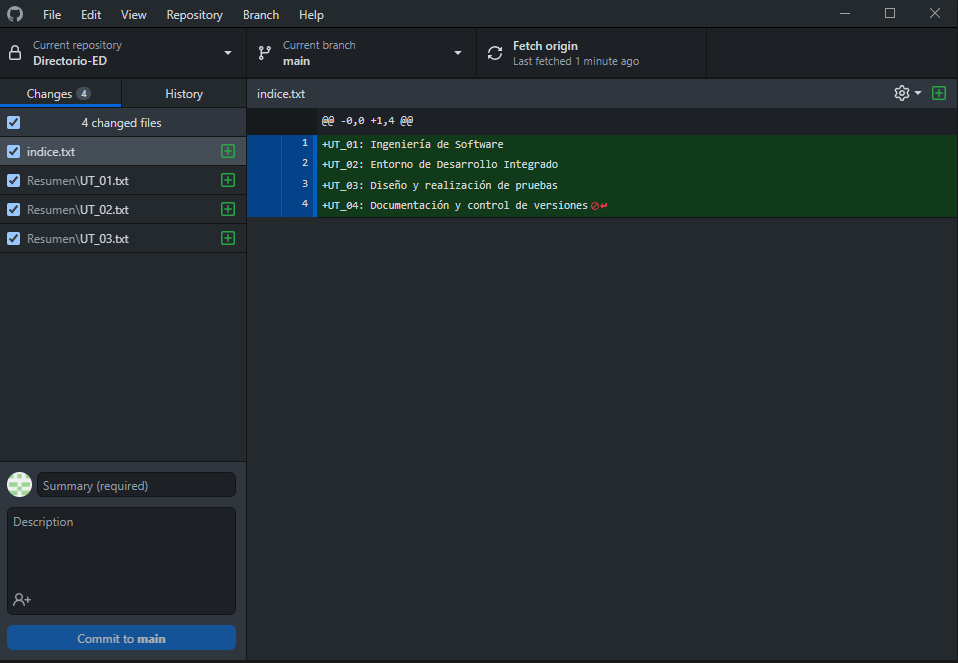
Elimino de nuevo la última línea del fichero indice.txt



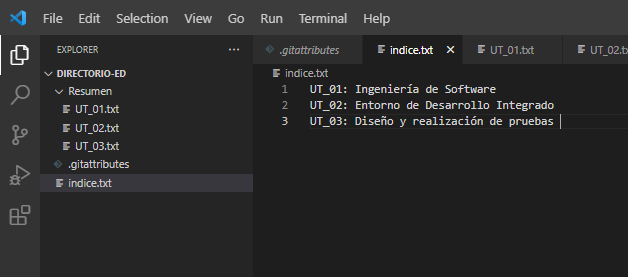
Compruebo el estado del repositorio:



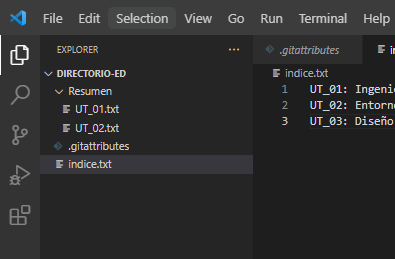
Vuelvo a la versión anterior y muestro los resultados:



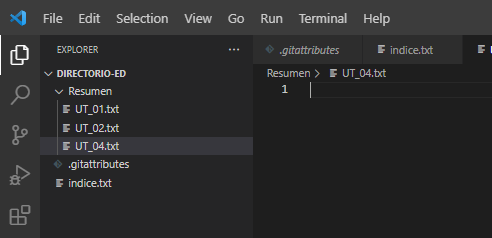
Elimino de nuevo la última línea del fichero indice.txt



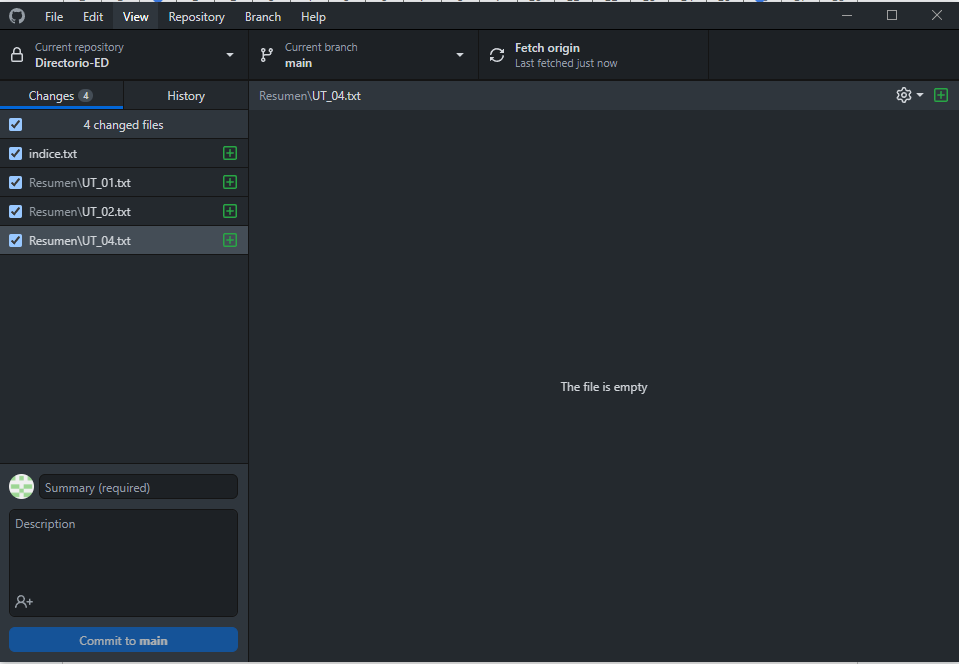
Elimino el fichero UT\_O3.txt



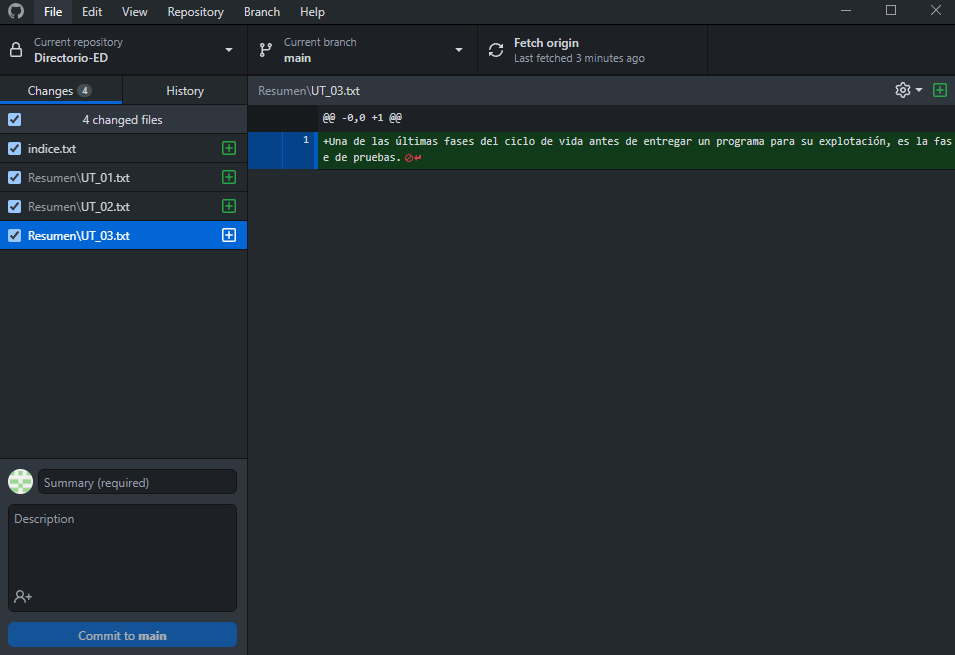
Creo un nuevo fichero vacio en la carpeta Resumen llamado UT\_04.txt



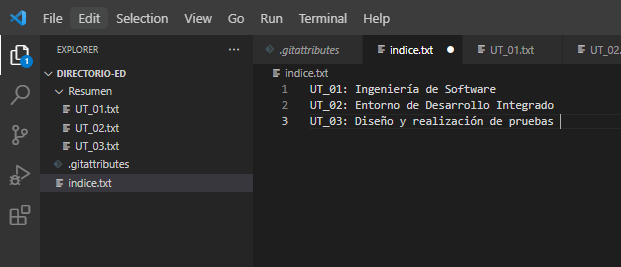
Compruebo el estado del repositorio:



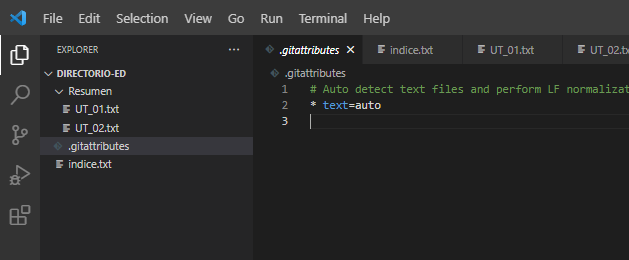
Deshacer los cambios anteriores y mostrar el resultado:



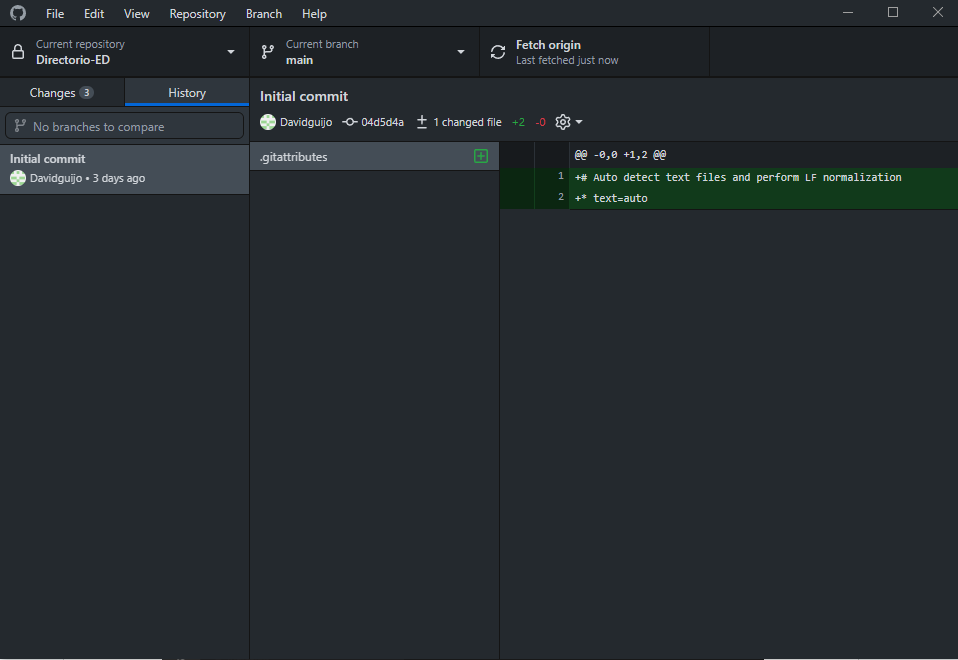
Elimino la última línea del fichero indice.txt:



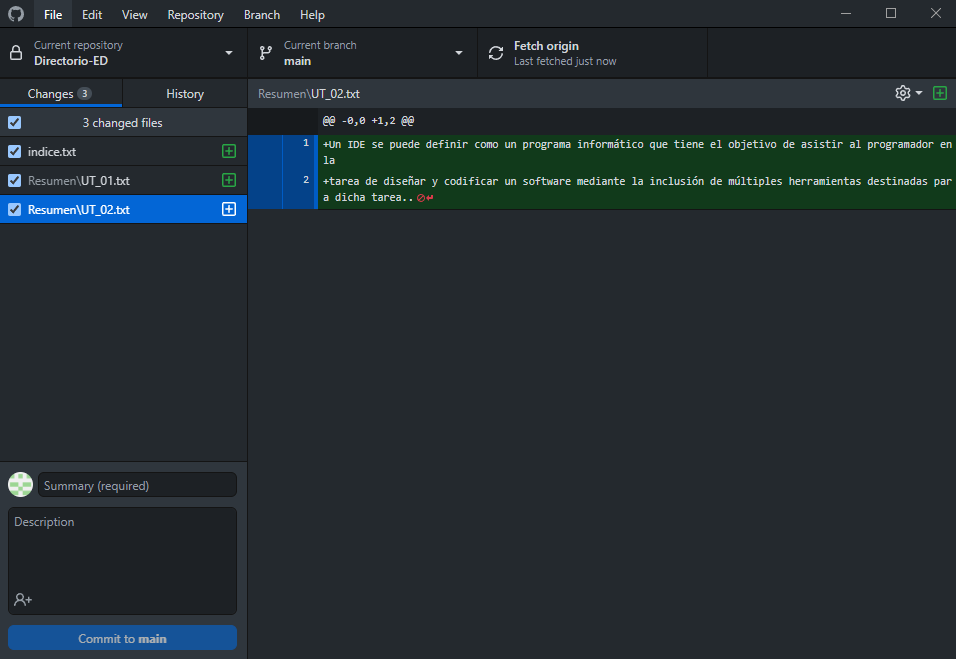
Elimino el fichero UT\_03.txt:



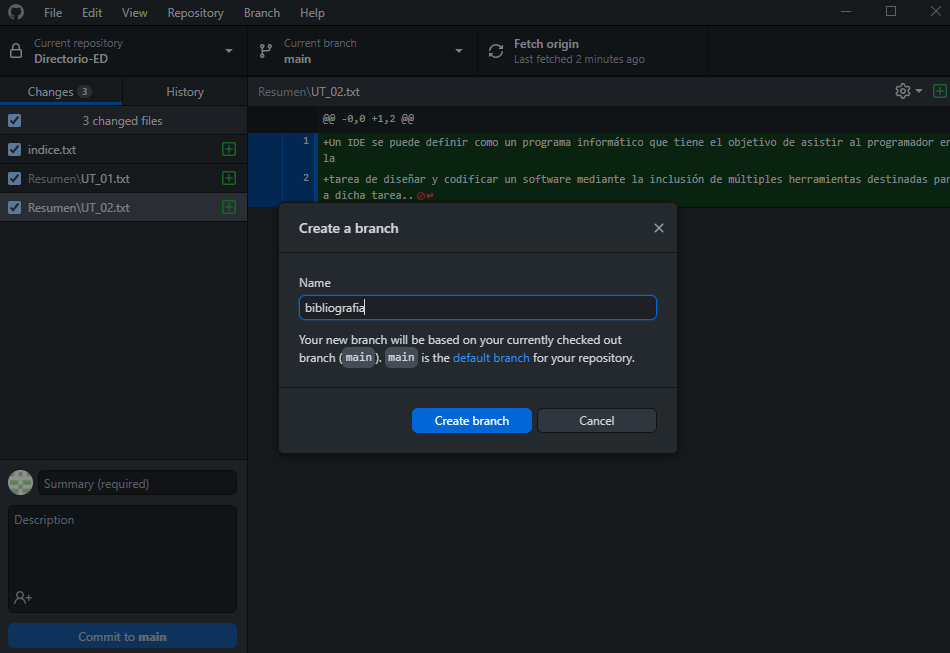
Compruebo el historial del repositorio:



Compruebo el estado del repositorio:

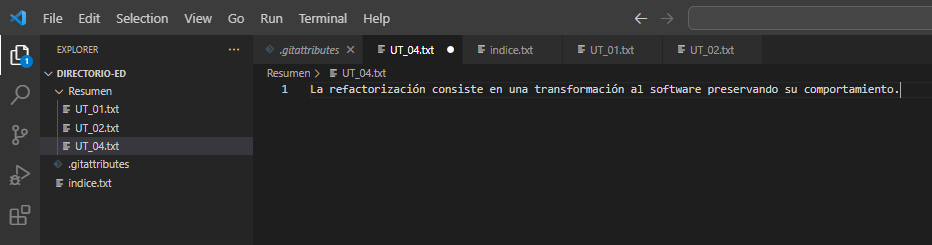


Creo una rama llamada bibliografía

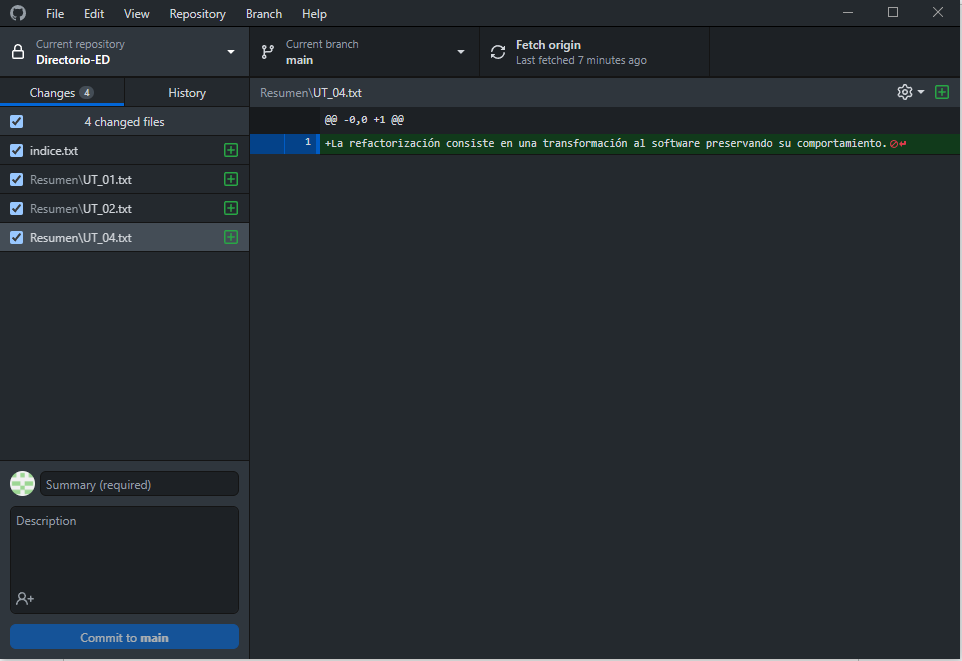


Creo en el main el fichero UT\_04.txt con el contenido:

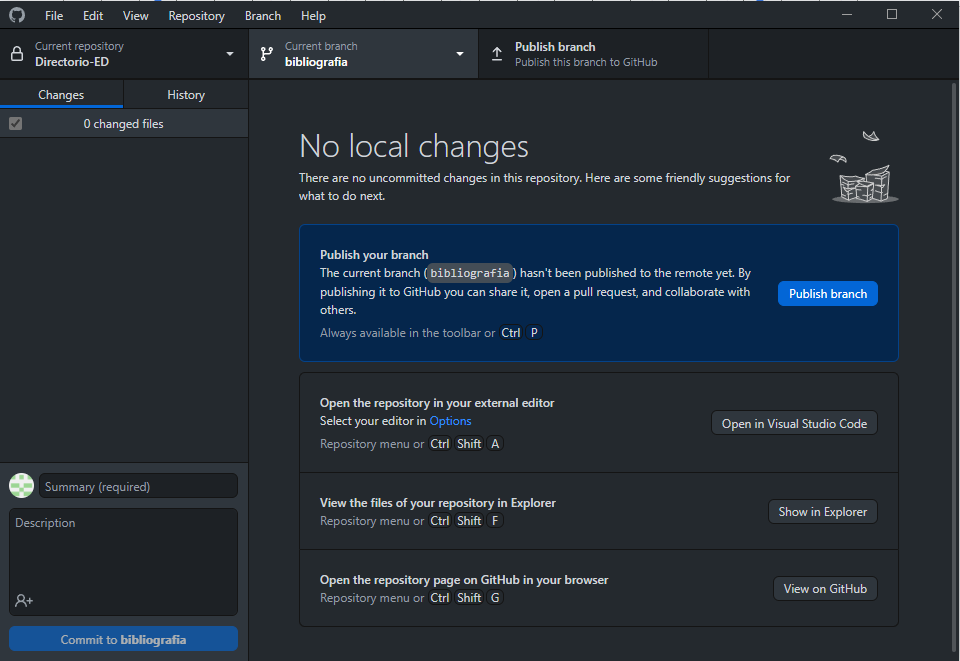
“La refactorización consiste en una transformación al software preservando su comportamiento.”



Muestro el estado del repositorio:

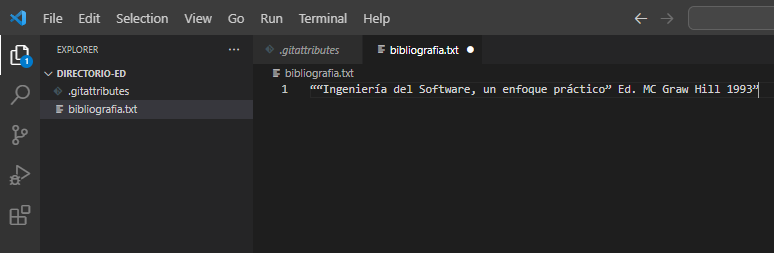


Cambio a la rama bibliografia:

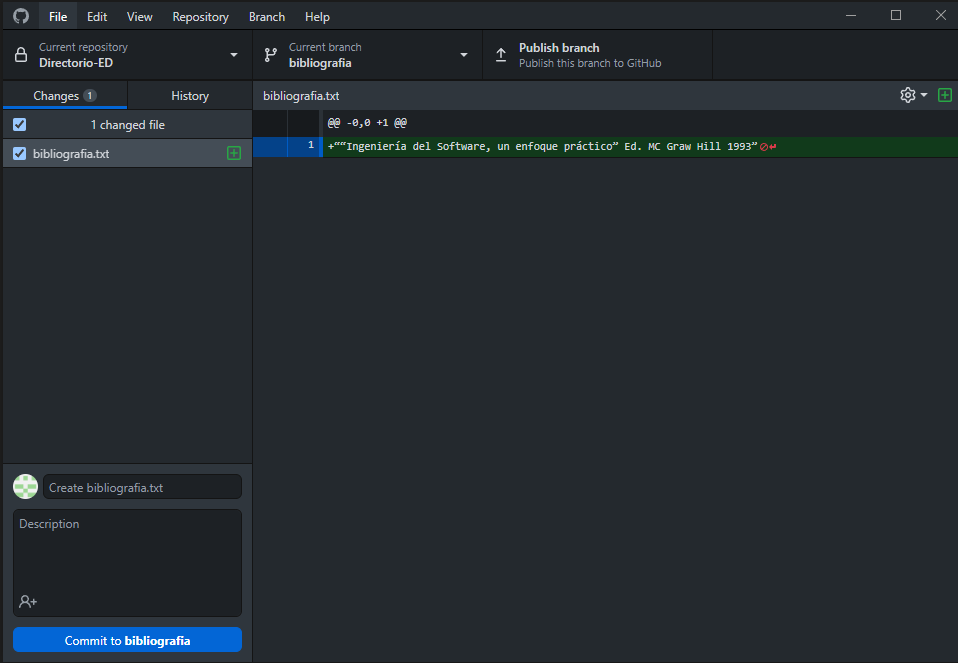


Creo el fichero bibliografia.txt con el siguiente contenido:

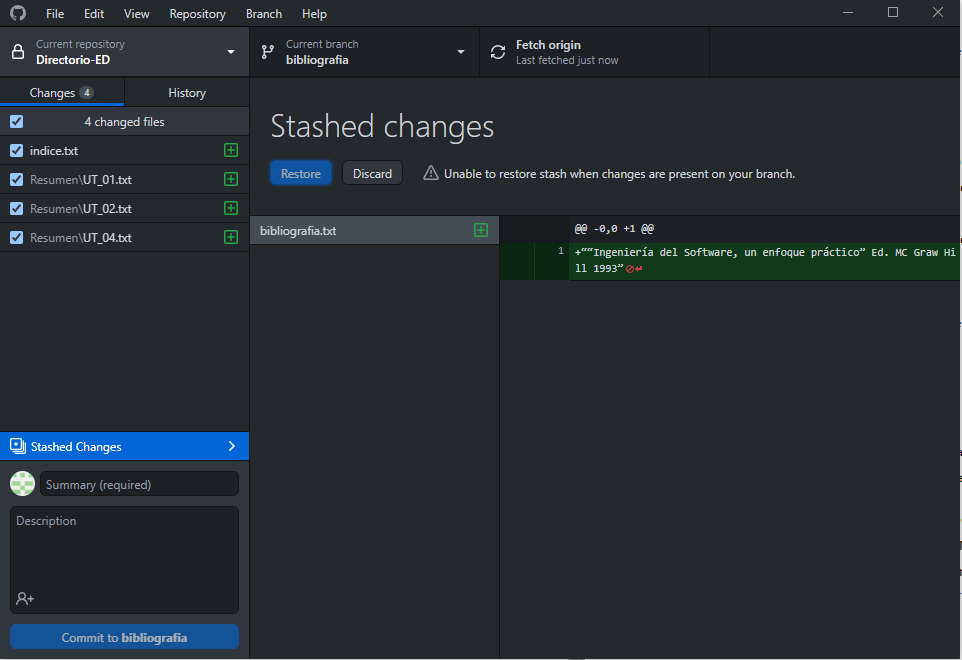
““Ingeniería del Software, un enfoque práctico” Ed. MC Graw Hill 1993”



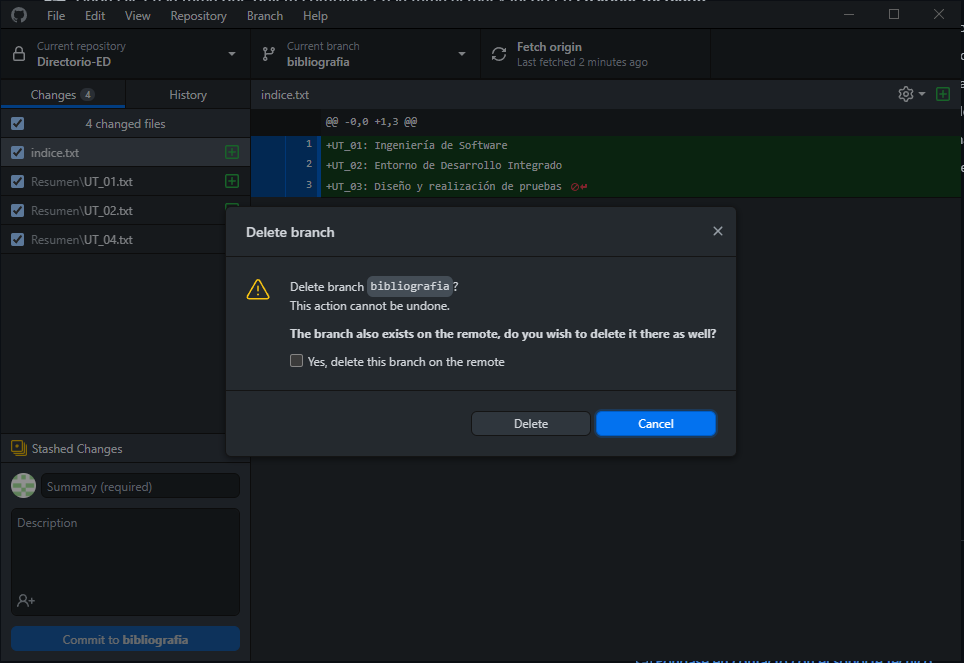
Muestro los cambios del repositorio:



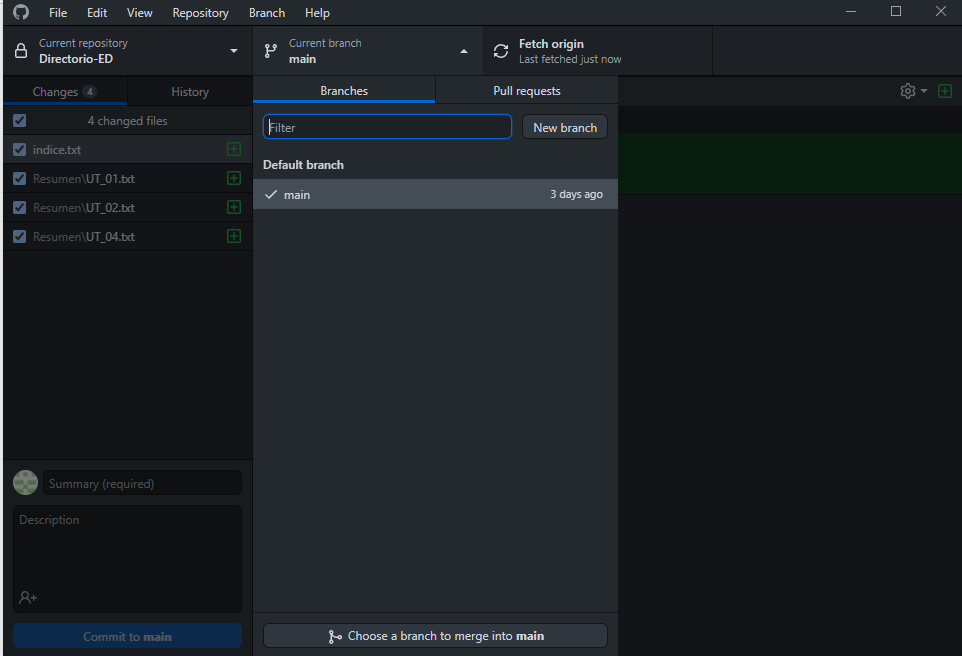
Fusiono las ramas:



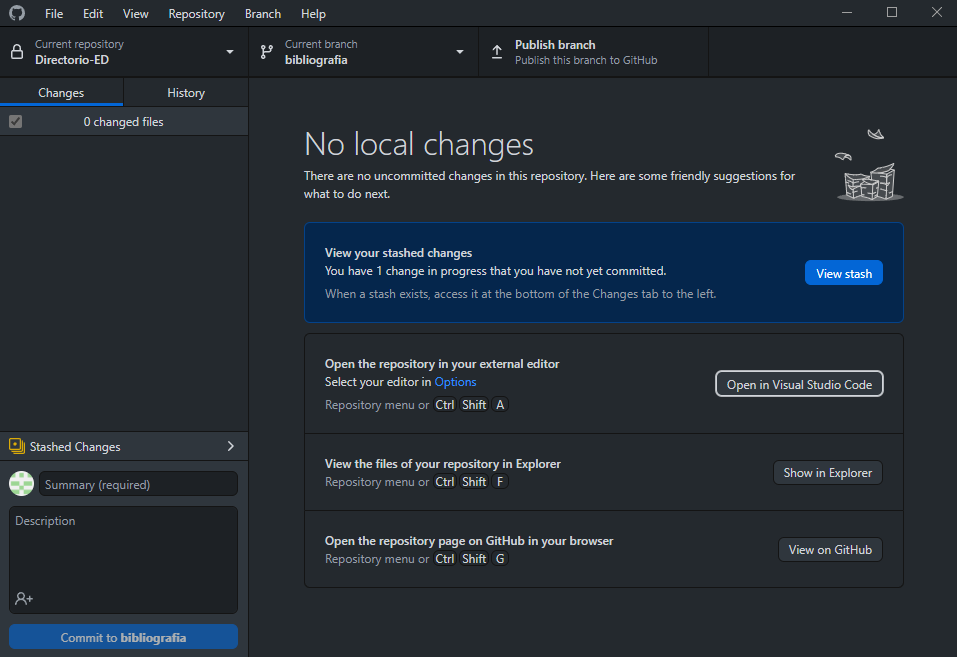
Elimino la rama bibliografia:



Compruebo el estado del repositorio:



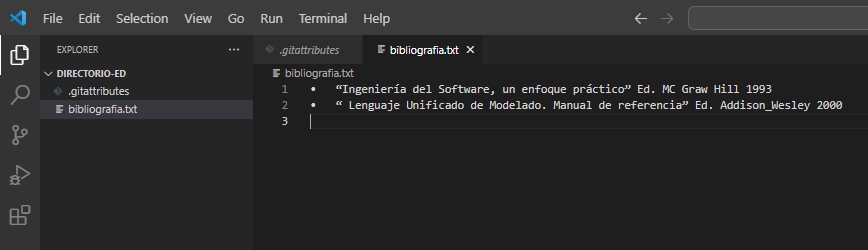
Creo de nuevo la rama bibliografia:



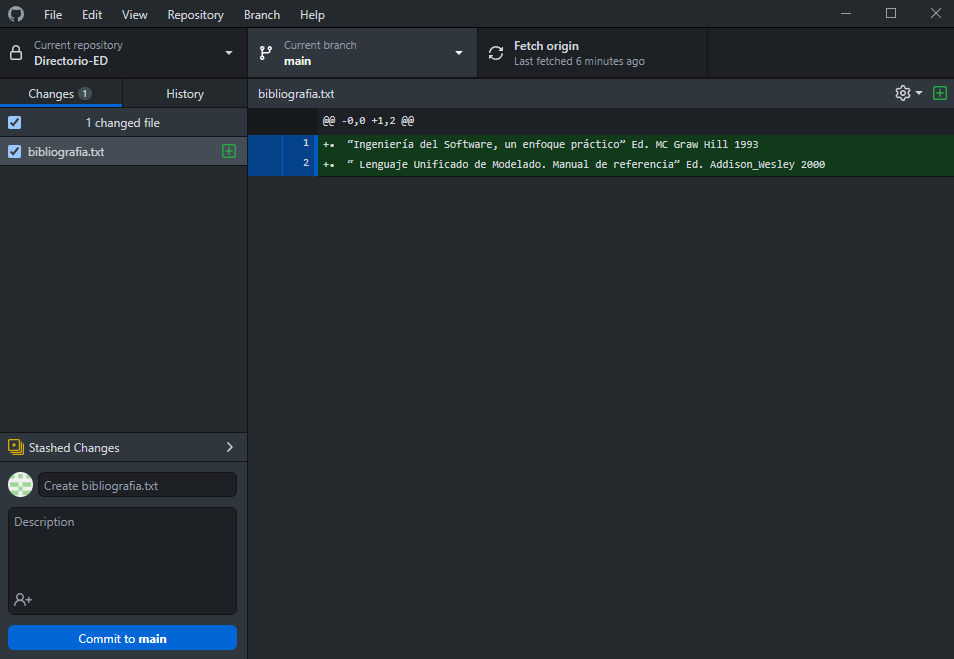
Creo en la rama bibliografia un fichero (bibliografia.txt) con el contenido:

· “Ingeniería del Software, un enfoque práctico” Ed. MC Graw Hill 1993

· “ Lenguaje Unificado de Modelado. Manual de referencia” Ed. Addison\_Wesley 2000



Cambio a la rama main:

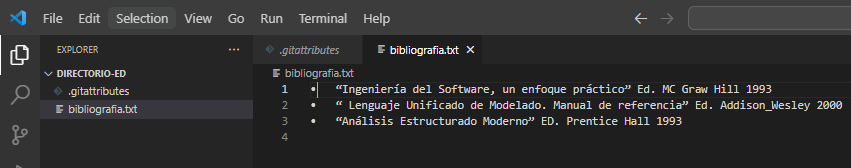


Cambio el fichero bibliografia.txt para que contenga:

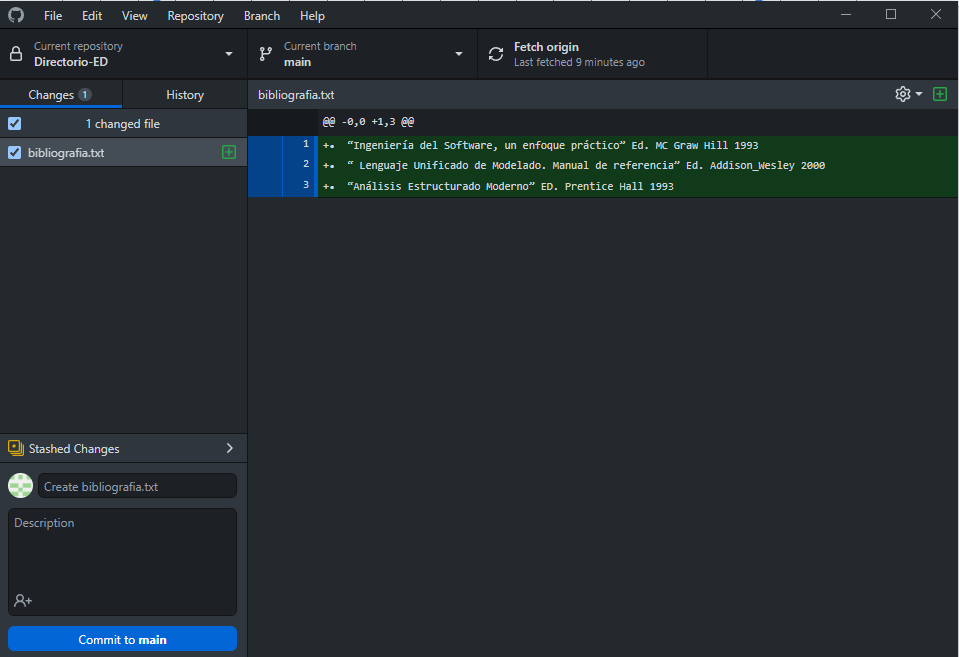
· “Ingeniería del Software, un enfoque práctico” Ed. MC Graw Hill 1993

· “ Lenguaje Unificado de Modelado. Manual de referencia” Ed. Addison\_Wesley 2000

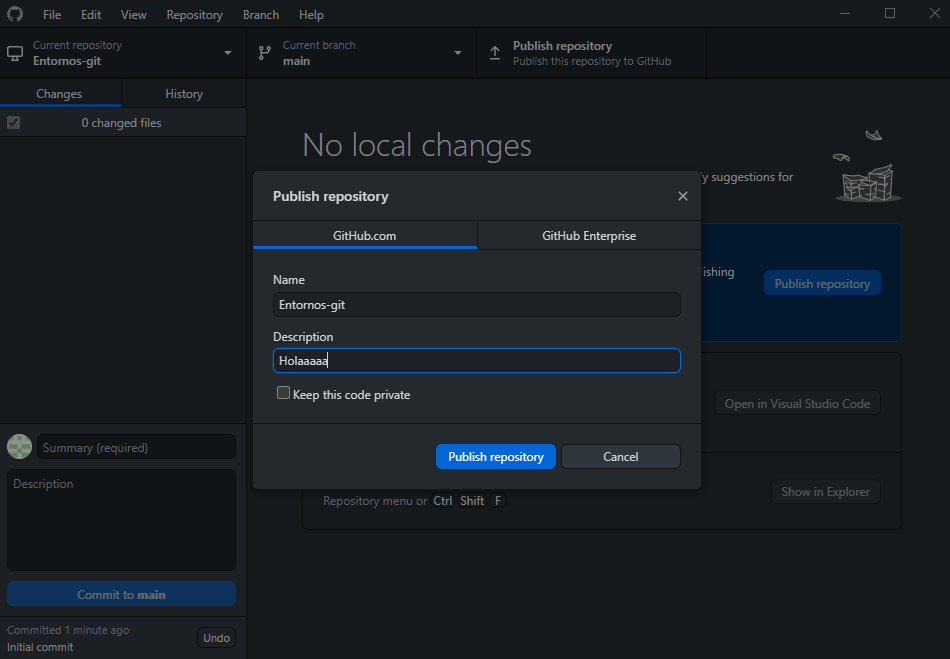
· “Análisis Estructurado Moderno” ED. Prentice Hall 1993



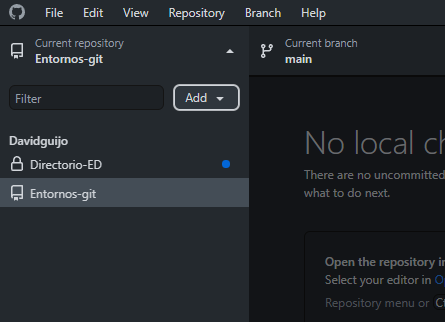
Compruebo el estado del repositorio:



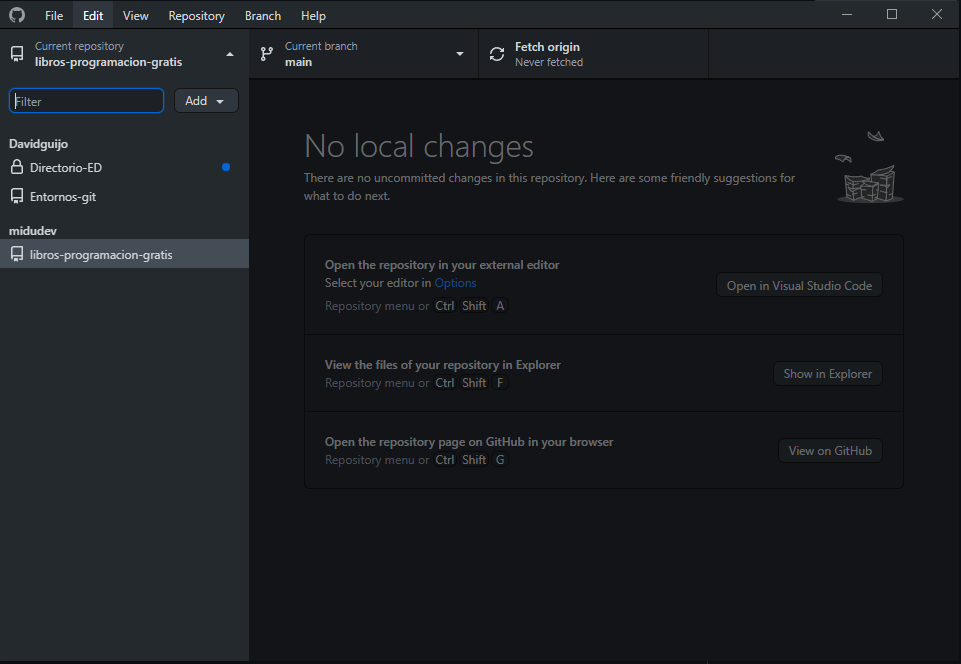
Creo un repositorio con el nombre Entornos-git



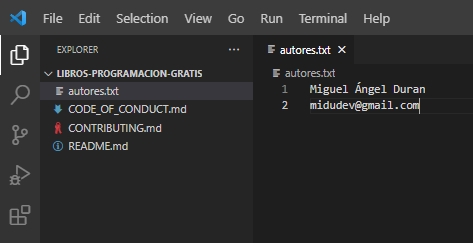
Muestro mis repositorios:



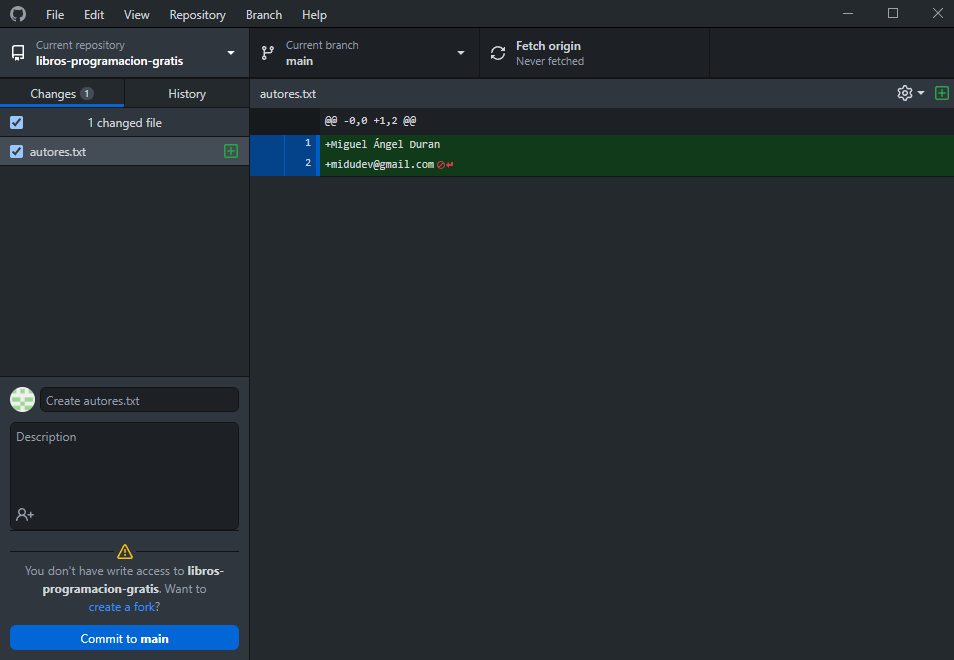
Clono un repositorio de internet:



Creo un fichero llamado autores.txt e introduzco su nombre y correo electrónico:



Muestro los cambios de repositorio:



**CONCLUSIÓN**

En conclusión GitHub es una herramienta super útil que nos permite gestionar proyectos y sus versiones de código.

GitHub es usado mundialmente por multitud de desarrolladores para almacenar sus trabajos, haciendo que los programadores tengan la posibilidad de cooperar entre ellos