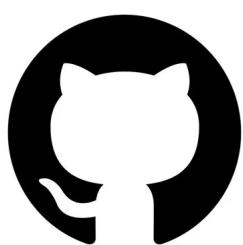
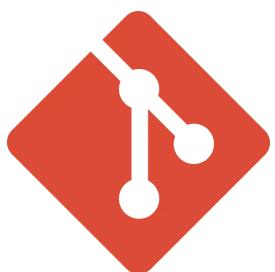


Instalación e introducción a Git/GitHub por interfaz gráfica



Introducción.....	3
A tener en cuenta.....	3
Instalación de Git.....	5
Clonación de un repositorio.....	22
Commit y Push de un proyecto.....	31
Pull.....	42
Crear una rama.....	48
Cambiar entre ramas.....	51
Merge de ramas.....	55
Resolver conflictos.....	57
Crear Pull Request.....	63

Introducción

El objetivo de este documento es explicar de manera resumida y guiar al usuario en la instalación y usos básicos de Git/GitHub a través de la interfaz gráfica del entorno de desarrollo Visual Studio code.

A tener en cuenta

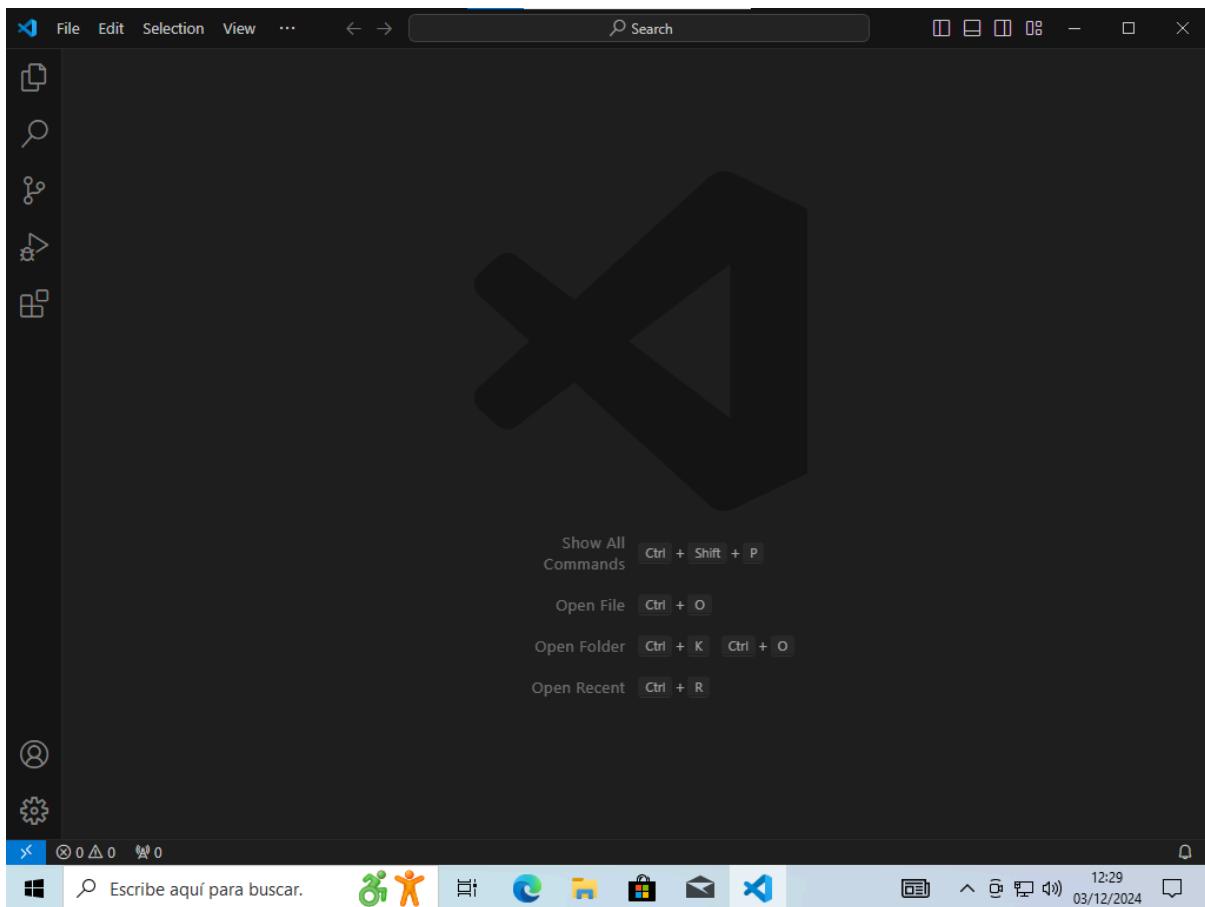
1. Los pasos y exemplificaciones de este documento están orientados para el software citado en la introducción. En caso de variar algún método o software puede implicar cambios en el proceso, aunque lo más probable es que para la mayoría de entornos de desarrollo o sistemas de control de versiones sea un proceso igual o similar.
2. Para el desarrollo de esta tarea se necesitará acceso a internet y descargar software externo (GIT) por lo que si ocurre algún problema en el proceso podría deberse a este punto o al punto número 1 del apartado “A tener en cuenta”.
3. Como punto de partida, se dará por hecho que...
 - a. El usuario conoce los conceptos y palabras claves relacionados con el control de versiones (pull, push, branch, merge, pull request).
 - b. El usuario tiene instalado el entorno de desarrollo Visual Studio Code sobre un sistema operativo Windows.
 - c. El usuario ya posee una cuenta de GitHub creada.

Instalación de Git

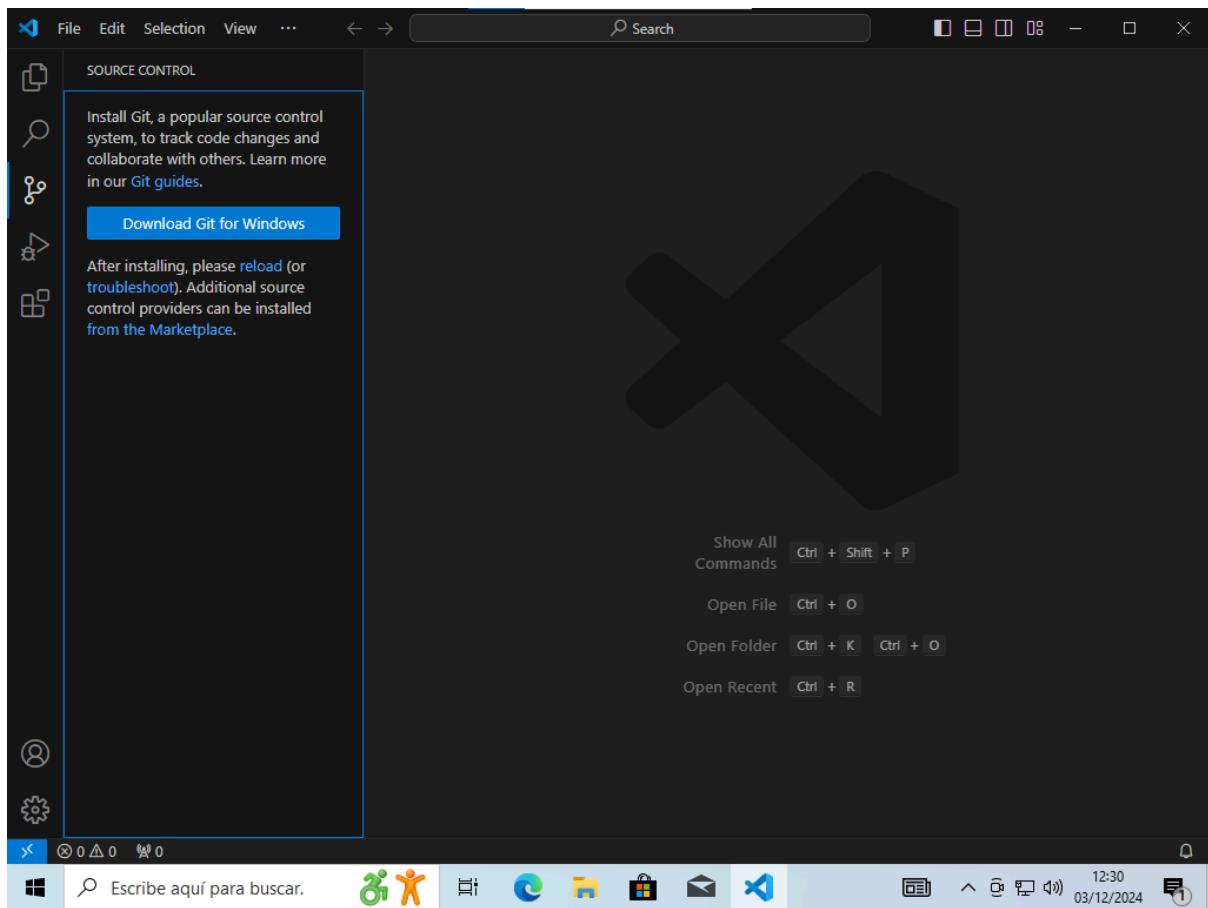
(Si ya tienes instalado Git en tu equipo, dirígete al punto “Clonación de un repositorio”)

Para instalar Git con la mayor simpleza posible seguiremos los siguientes pasos:

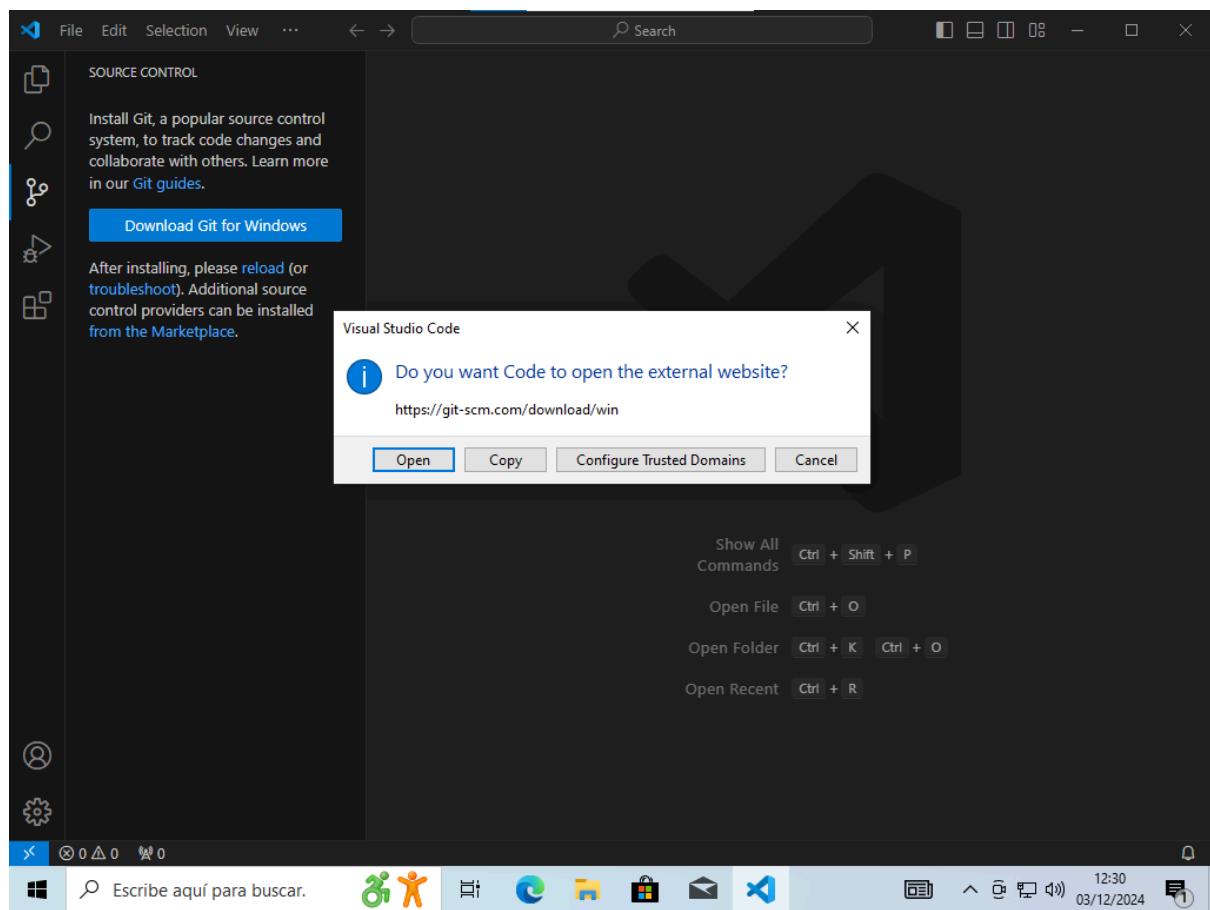
Abrir nuestro entorno de desarrollo Visual Studio Code.



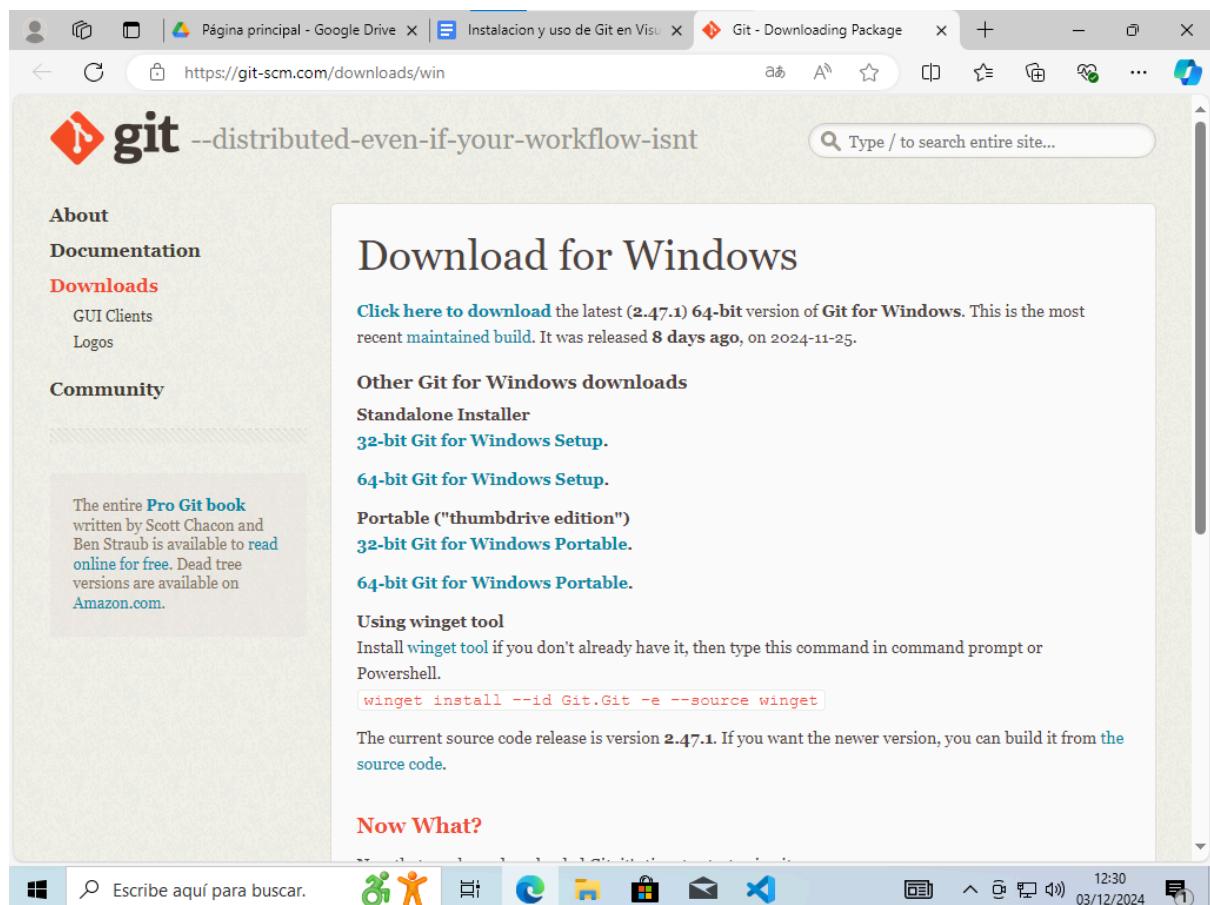
Nos dirigiremos al panel lateral, más específicamente a la 3^a opción enfocada en el control de versiones.



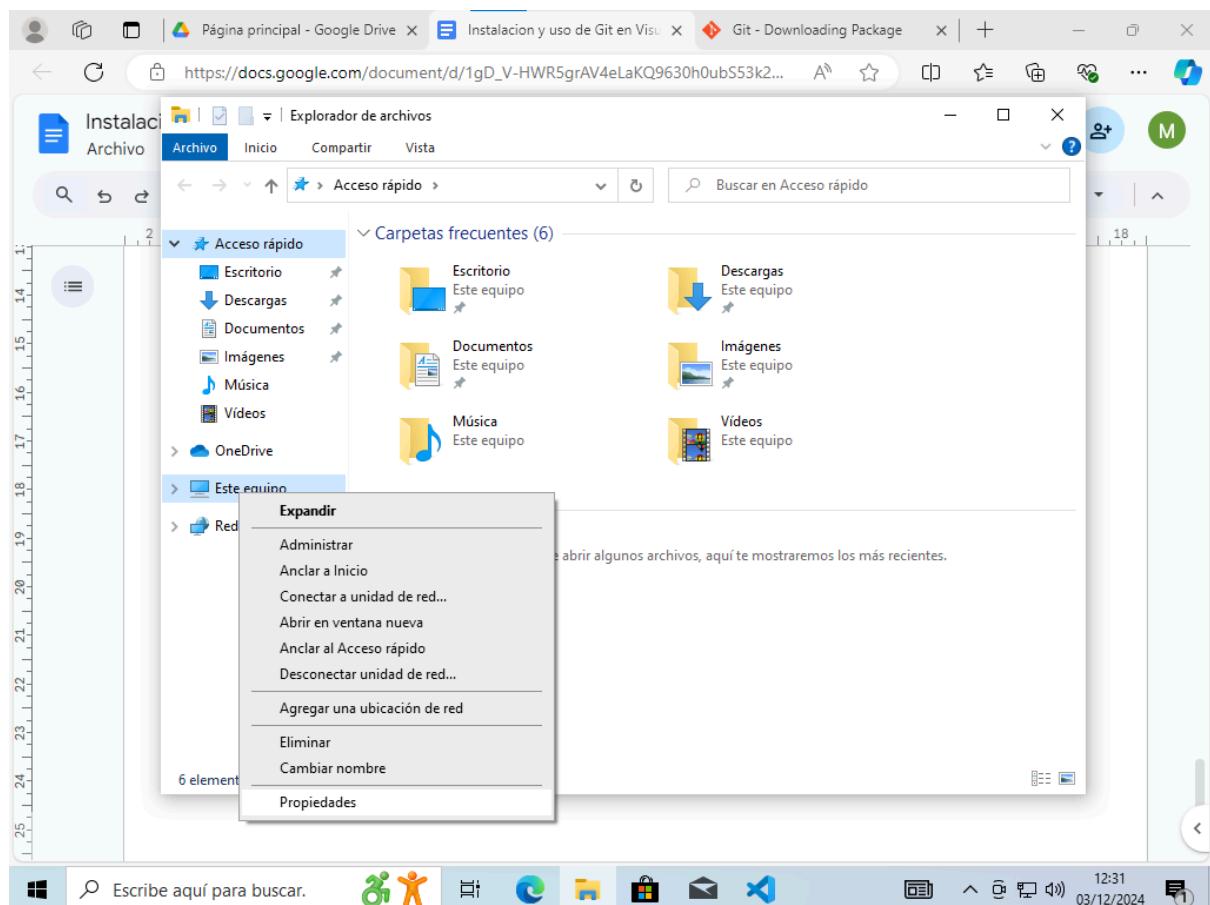
Al no detectarse ningún software apropiado para el control de versiones, el propio entorno de desarrollo nos proporciona un acceso a la página oficial de Git. Pulsaremos el botón azul “Download Git for Windows” / “Descargar Git para Windows”. Como confirmación de seguridad, nos pregunta si queremos abrir un enlace externo, le daremos a “Open” / “Abrir”.



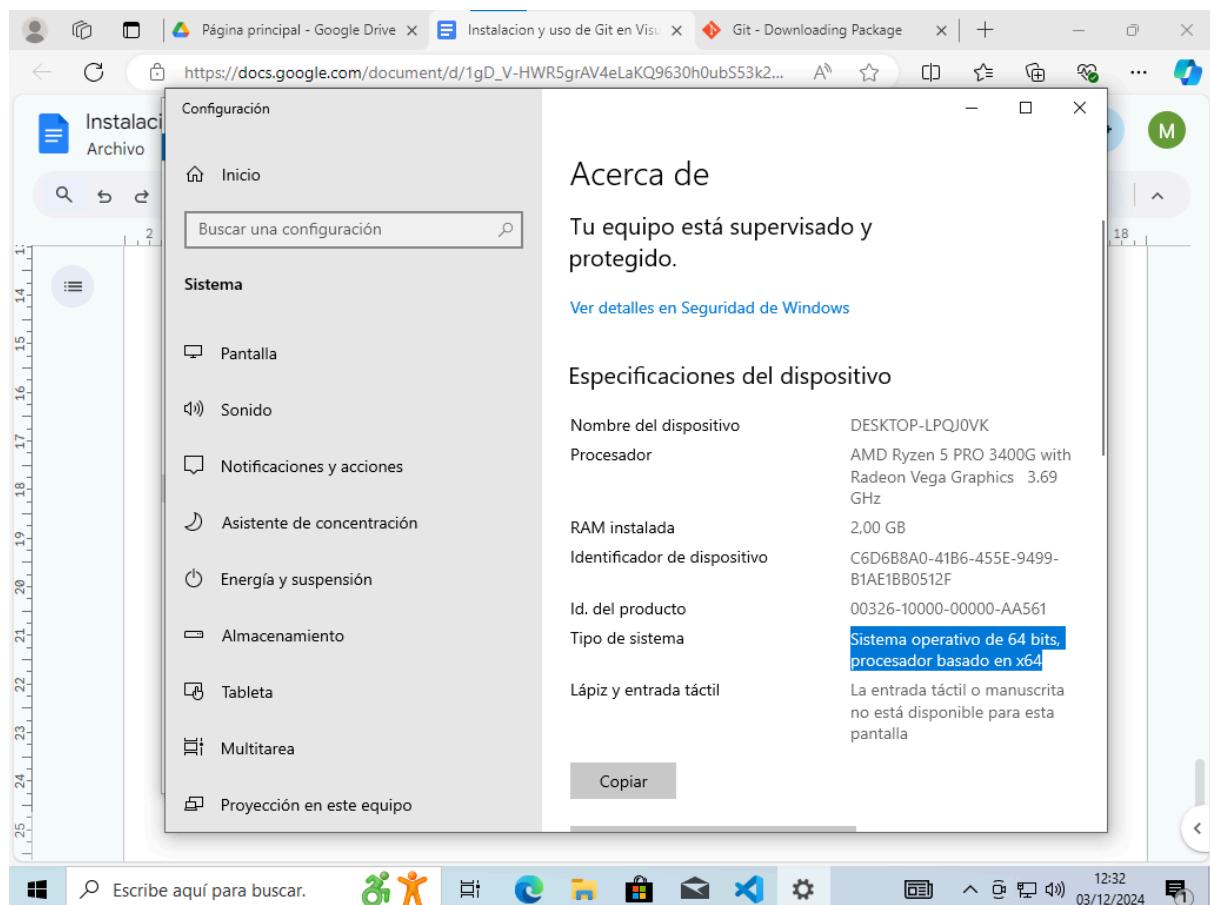
En nuestro navegador por defecto se nos abrirá la página oficial de Git.



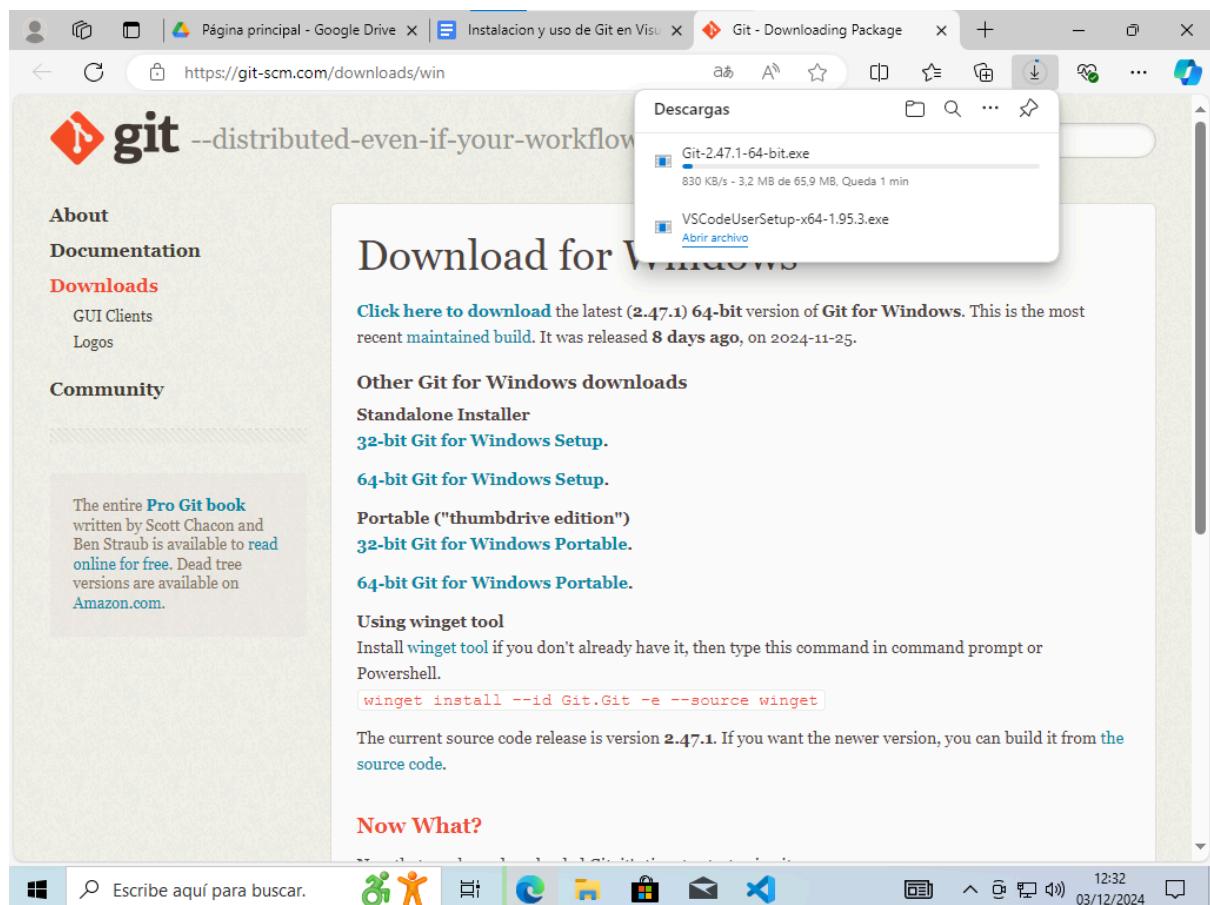
Para continuar debemos averiguar la arquitectura de nuestro procesador (x64 bits o x86/x32 bits). Para ello abrimos nuestro explorador de archivos, pulsaremos con click derecho sobre el ícono “Este equipo” ubicado en el panel lateral izquierdo y seleccionaremos la opción de “Propiedades” que, por lo general se ubica al final del menú contextual.



Seguidamente se abrirá la aplicación de configuración o una ventana específica de las propiedades del equipo (dependiendo de la versión de Windows). Aquí podremos ver nuestra arquitectura en el apartado “Tipo de sistema”.

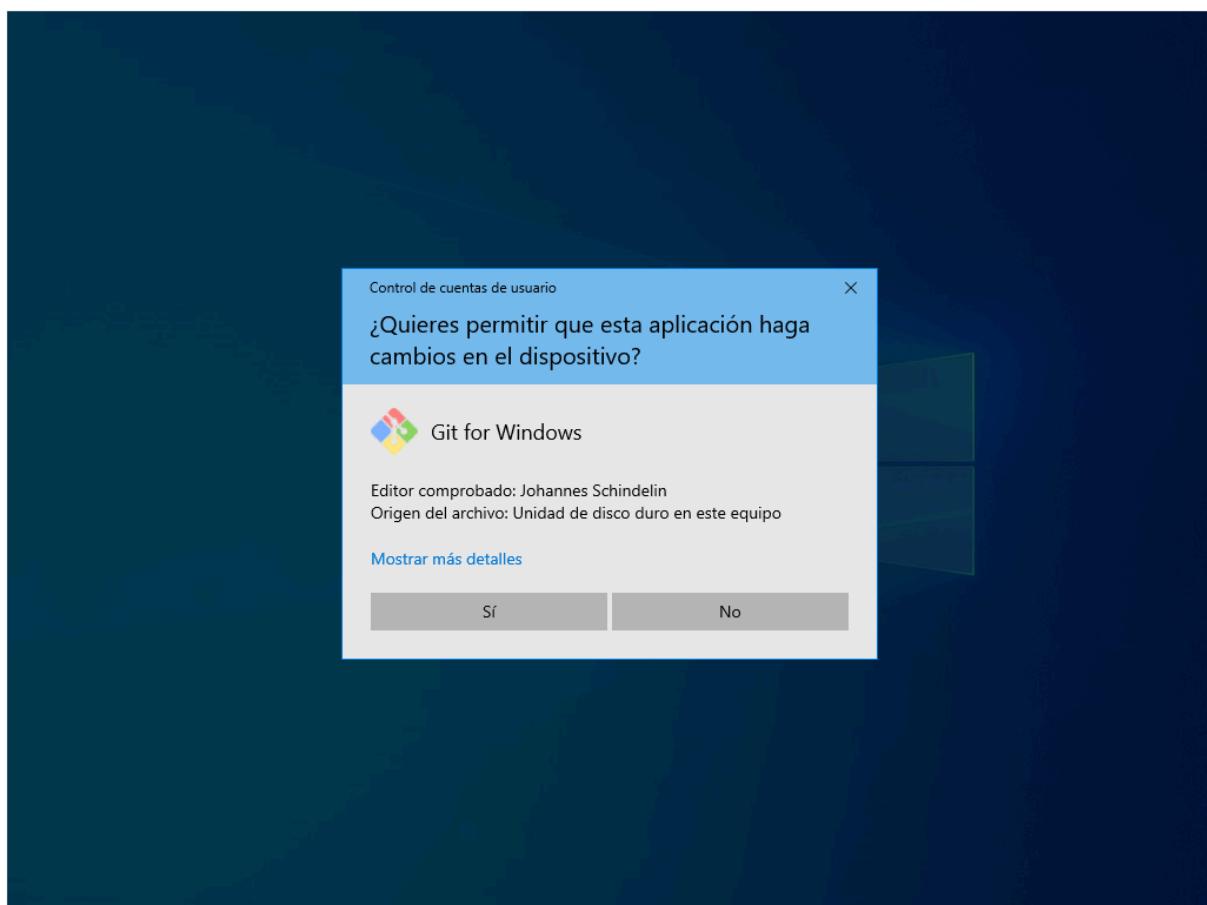


Conociendo la arquitectura de la CPU de nuestro equipo podremos escoger la versión que nos convenga (x64 bits = “64-bits Git for Windows Setup” o x32 bits / x86 bits = “32-bits Git for Windows Setup”). Al pulsar el enlace azul correspondiente se nos comenzará a descargar un ejecutable de Git.

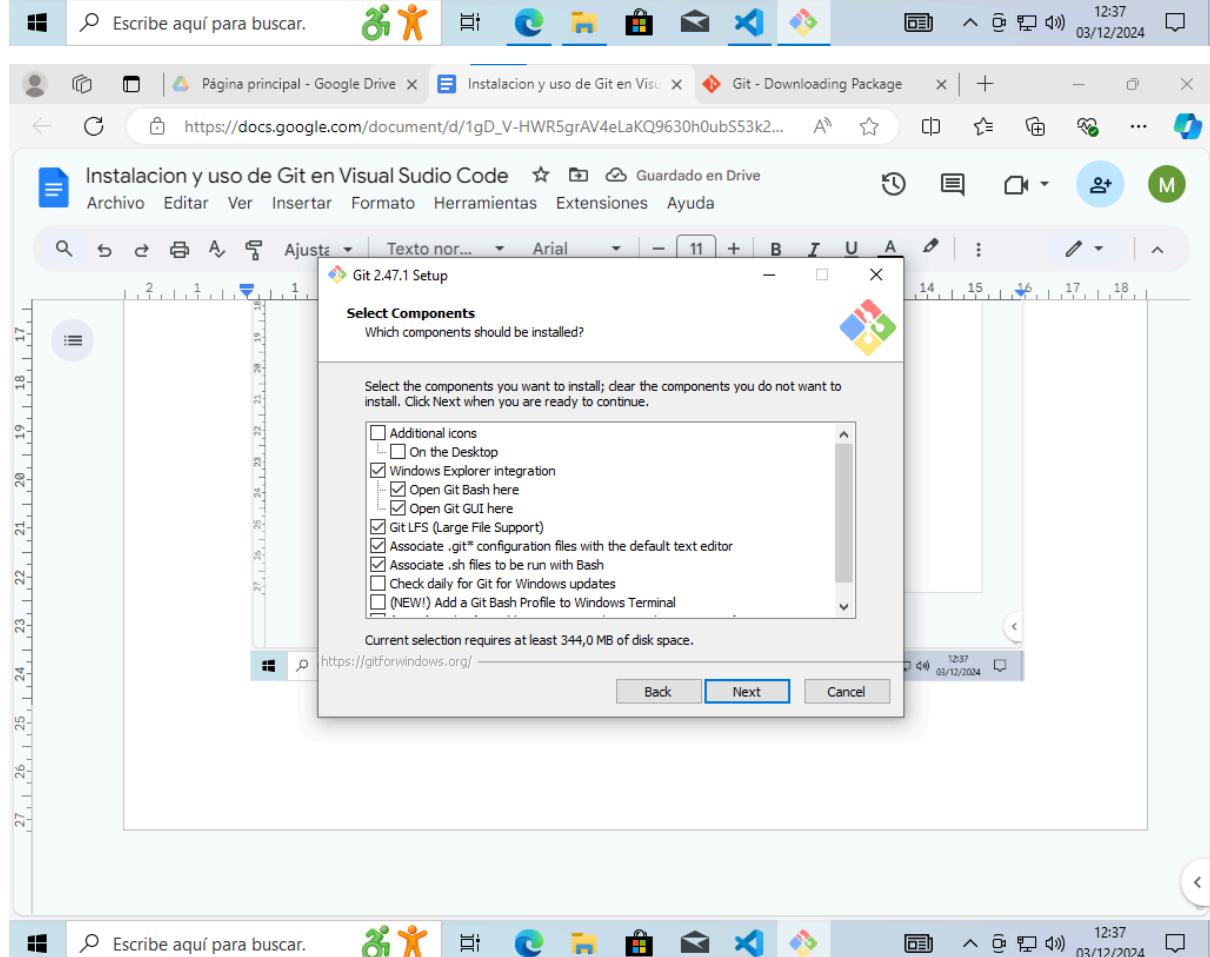
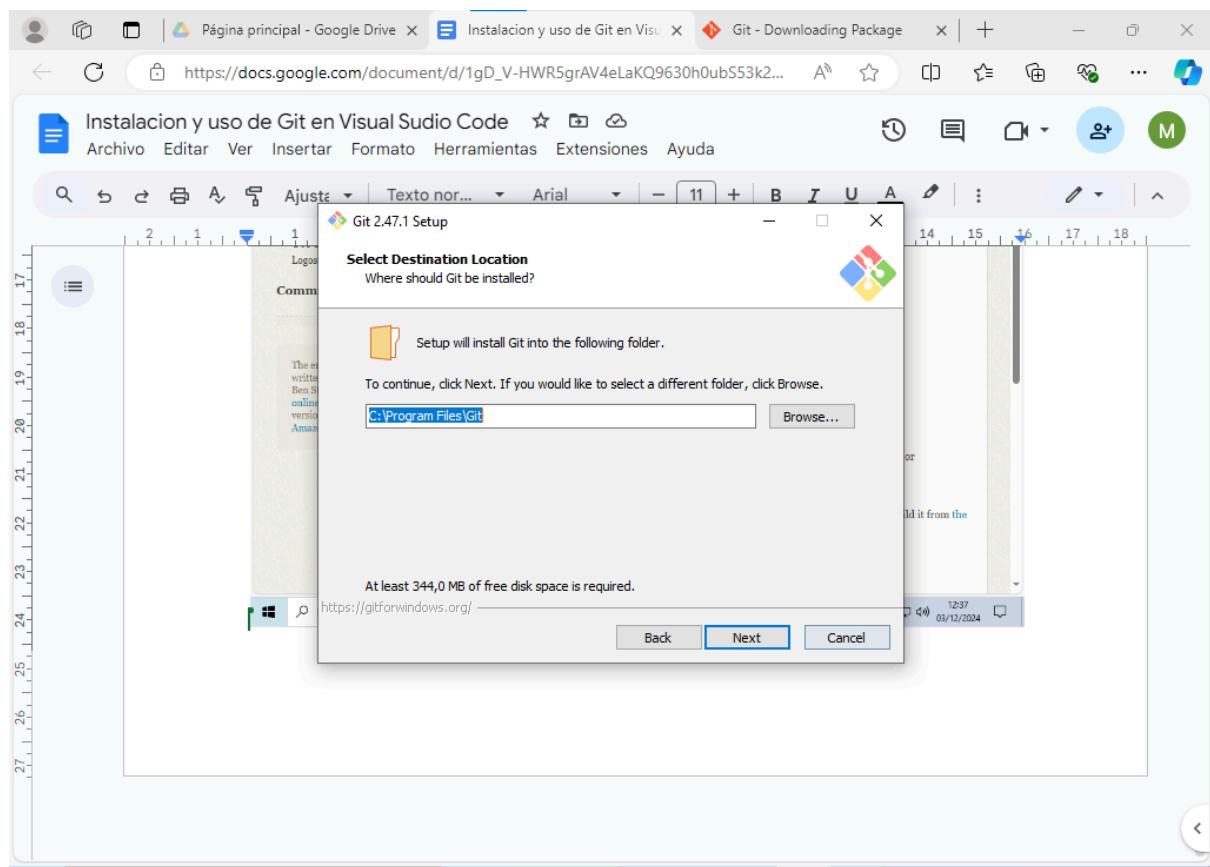


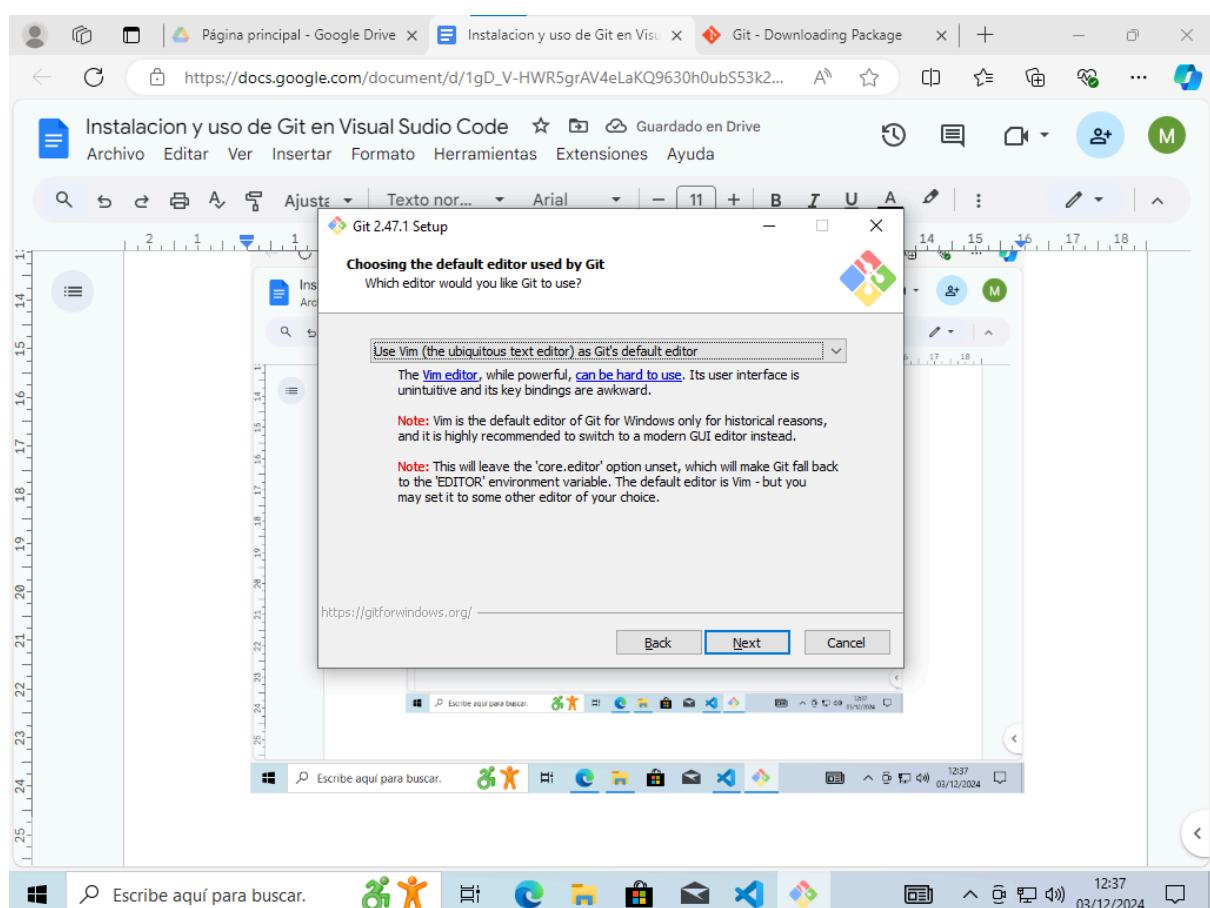
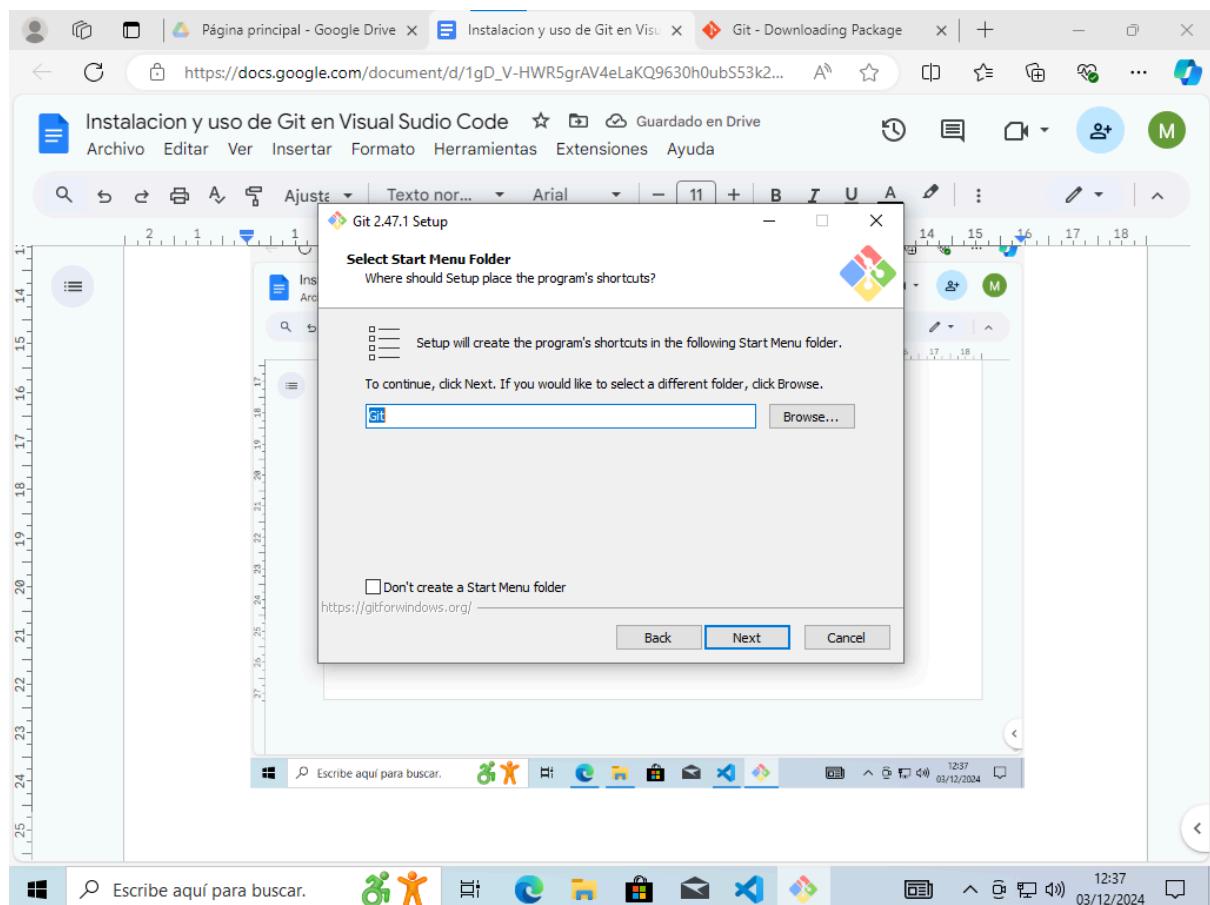
Una vez descargado el ejecutable lo abrimos para comenzar el proceso de instalación.

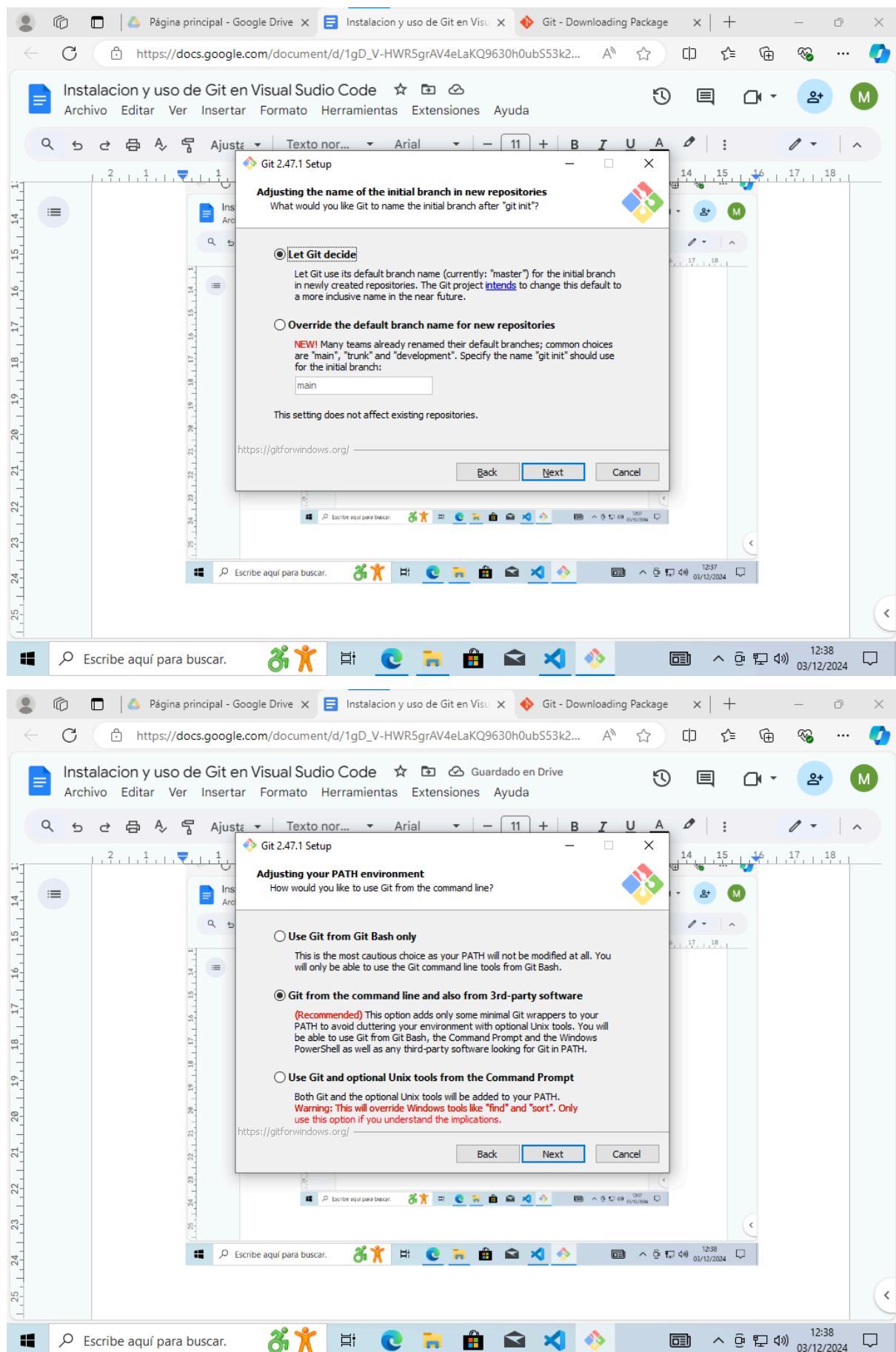
A continuación aparecerán las pantallas del proceso de instalación 1 por 1. Todos los parámetros se dejarán por defecto y simplemente pulsaremos el botón que nos permita avanzar en el proceso.

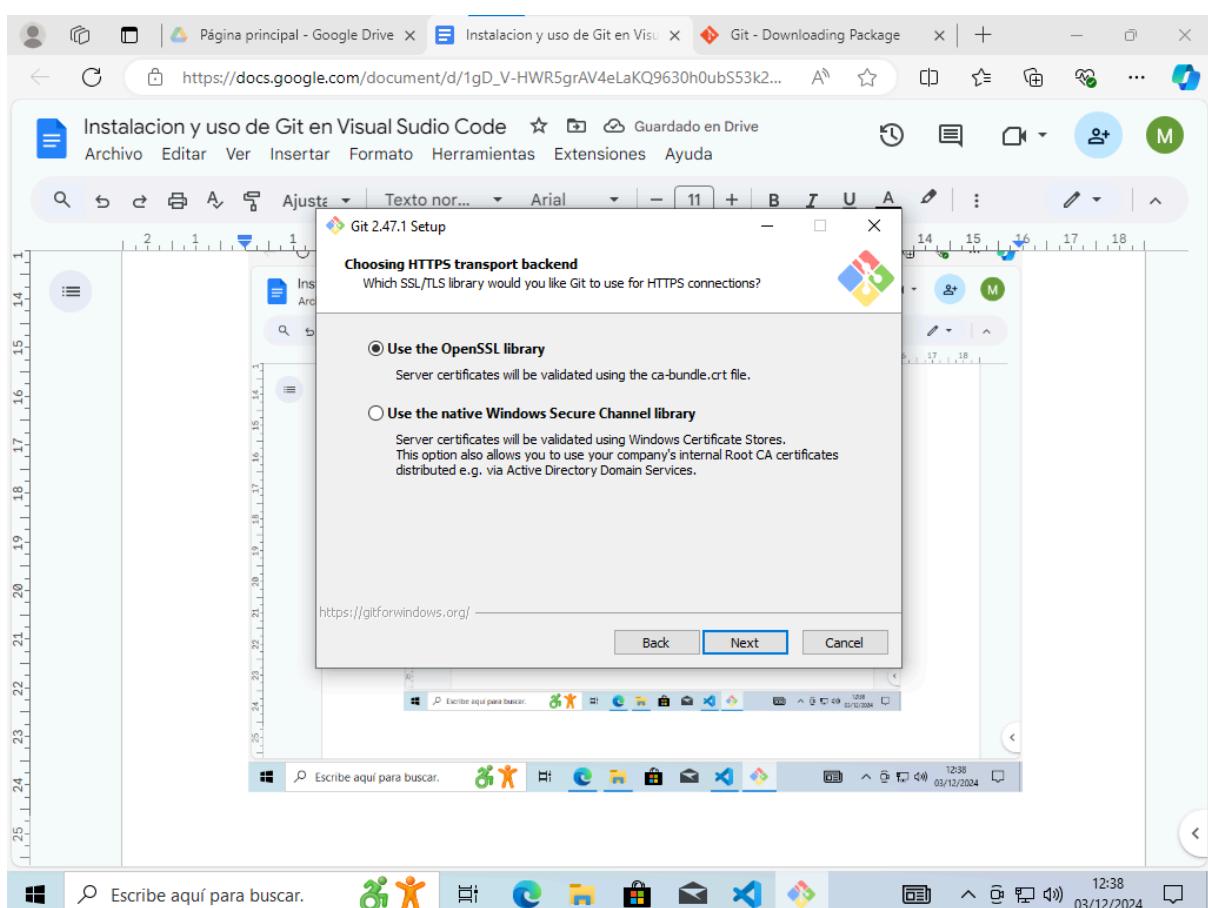
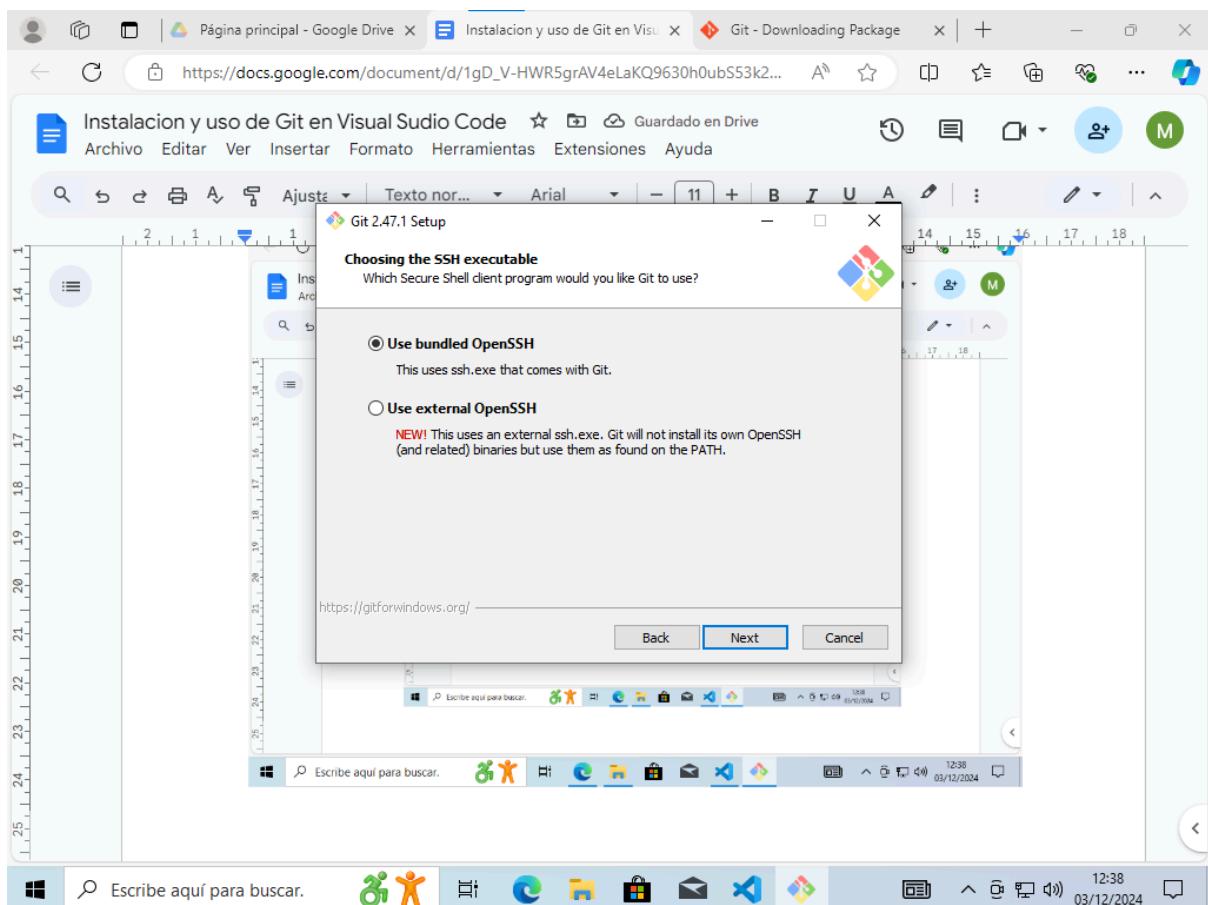


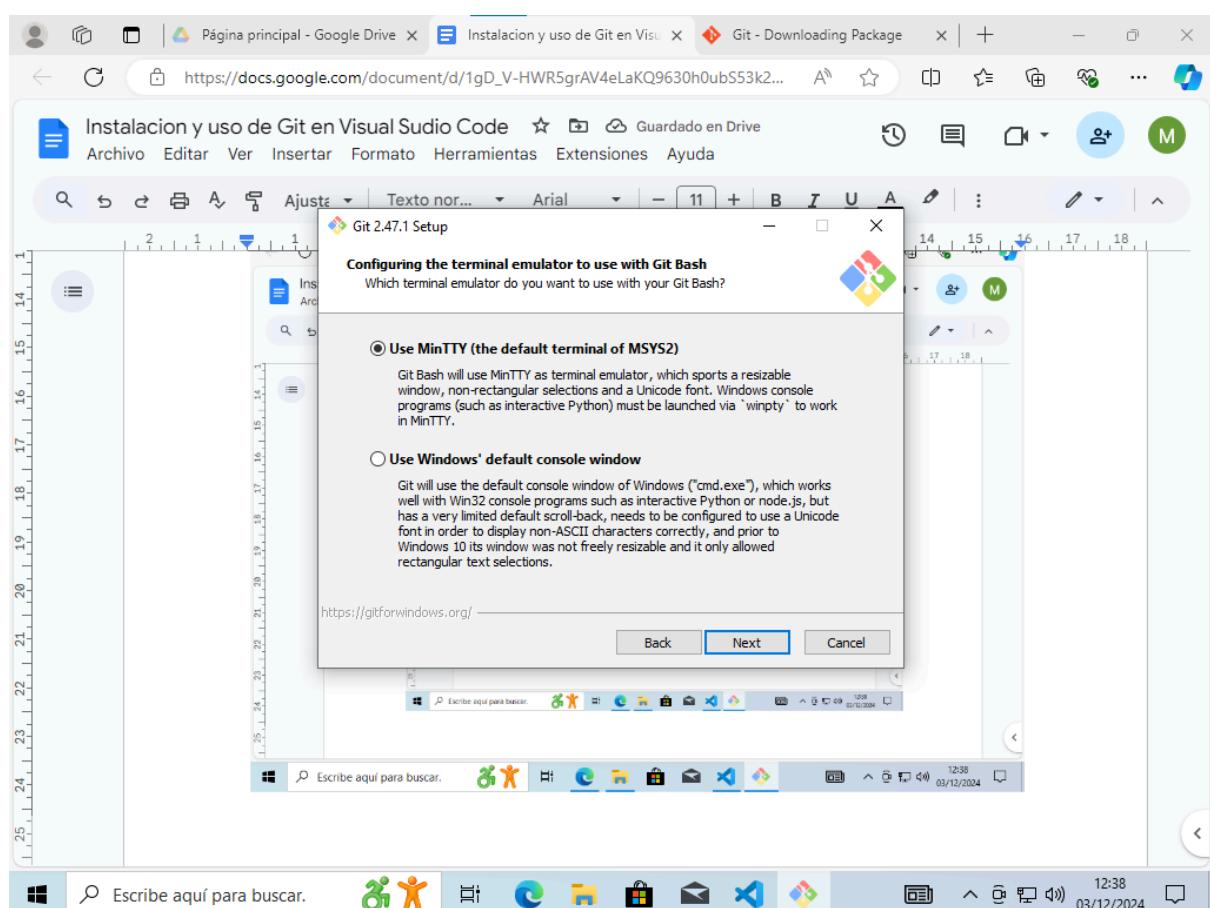
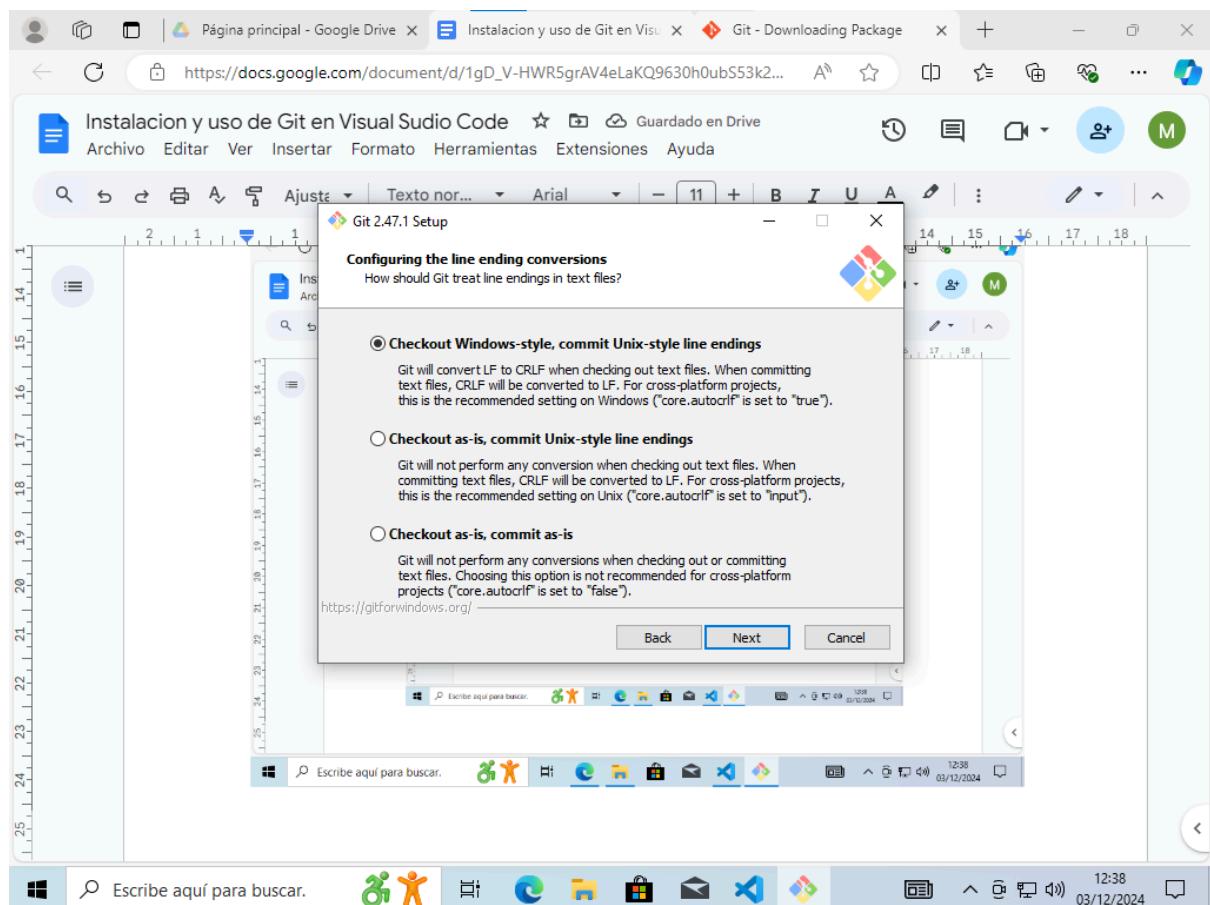
A screenshot of a Windows desktop. In the center, a "Git 2.47.1 Setup" window is open, displaying the "Information" screen of the "GNU General Public License". The license text is visible, mentioning "Version 2, June 1991" and the "Preamble". Above this window, a "Descargas" (Downloads) dialog box is open, showing a single item: "Git-2.47.1-64-bit.exe" with the option "Abrir archivo" (Open file). In the background, a web browser window is visible, showing the "git-scm.com/downloads/win" page. The browser's address bar shows the URL "https://git-scm.com/downloads/win". The taskbar at the bottom includes icons for Start, Search, File Explorer, Edge, File Manager, Mail, and Visual Studio Code, along with system status icons.

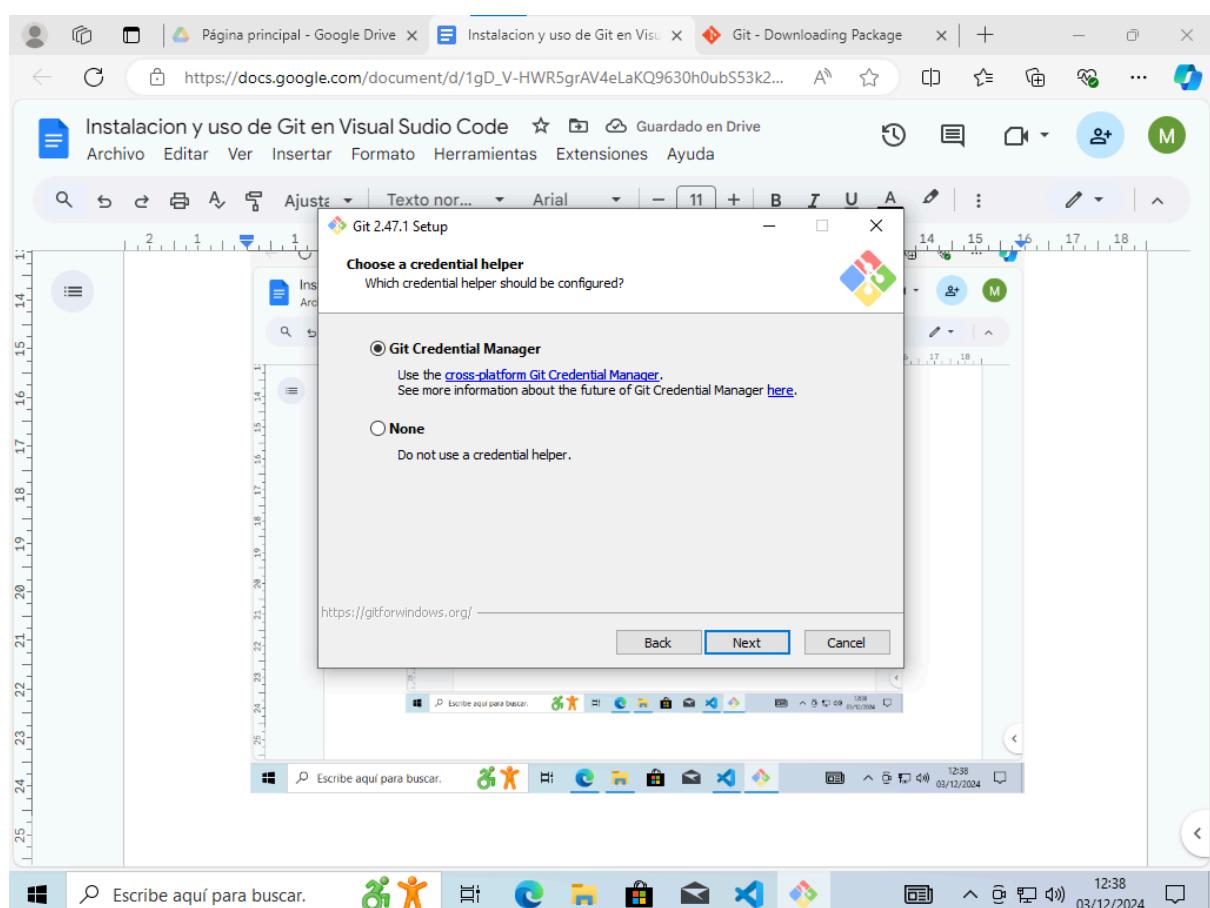
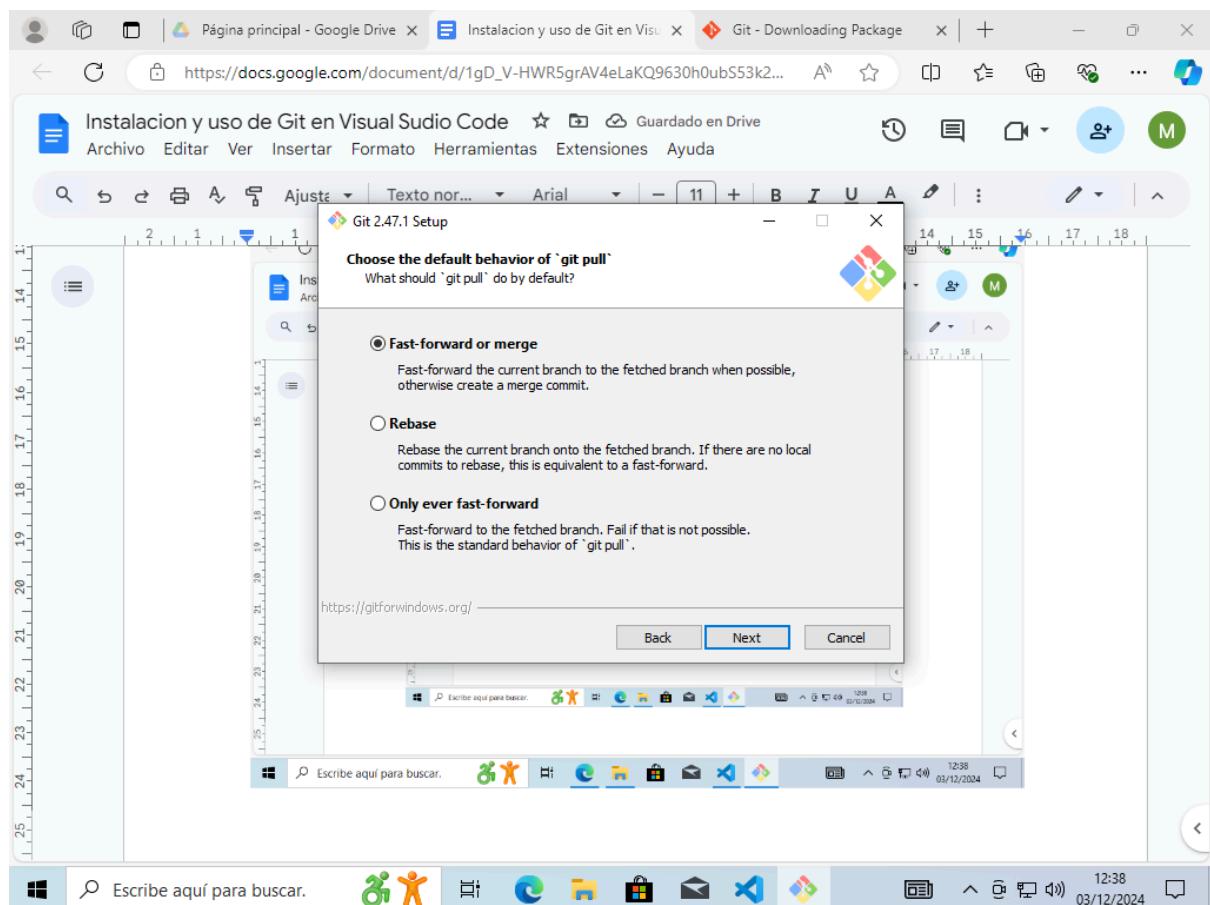


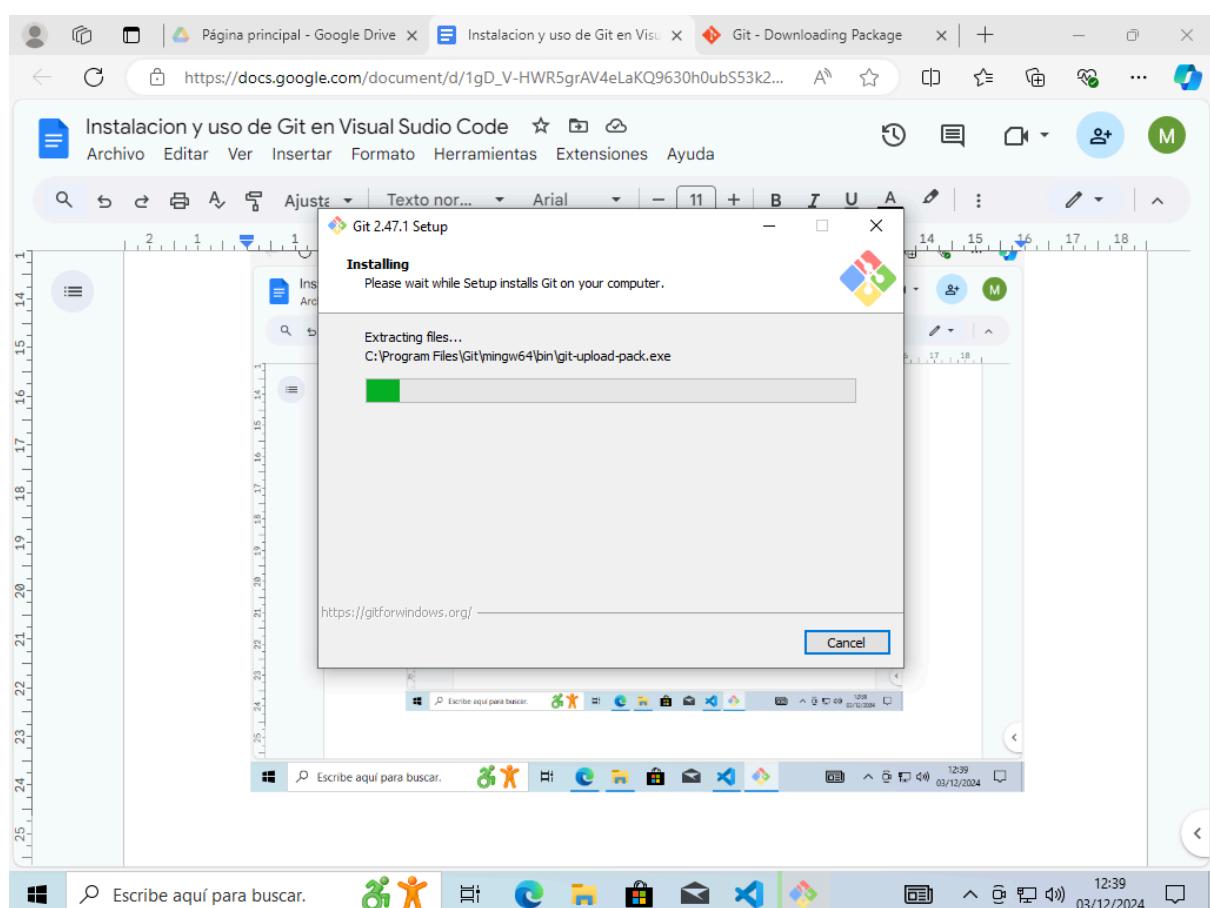
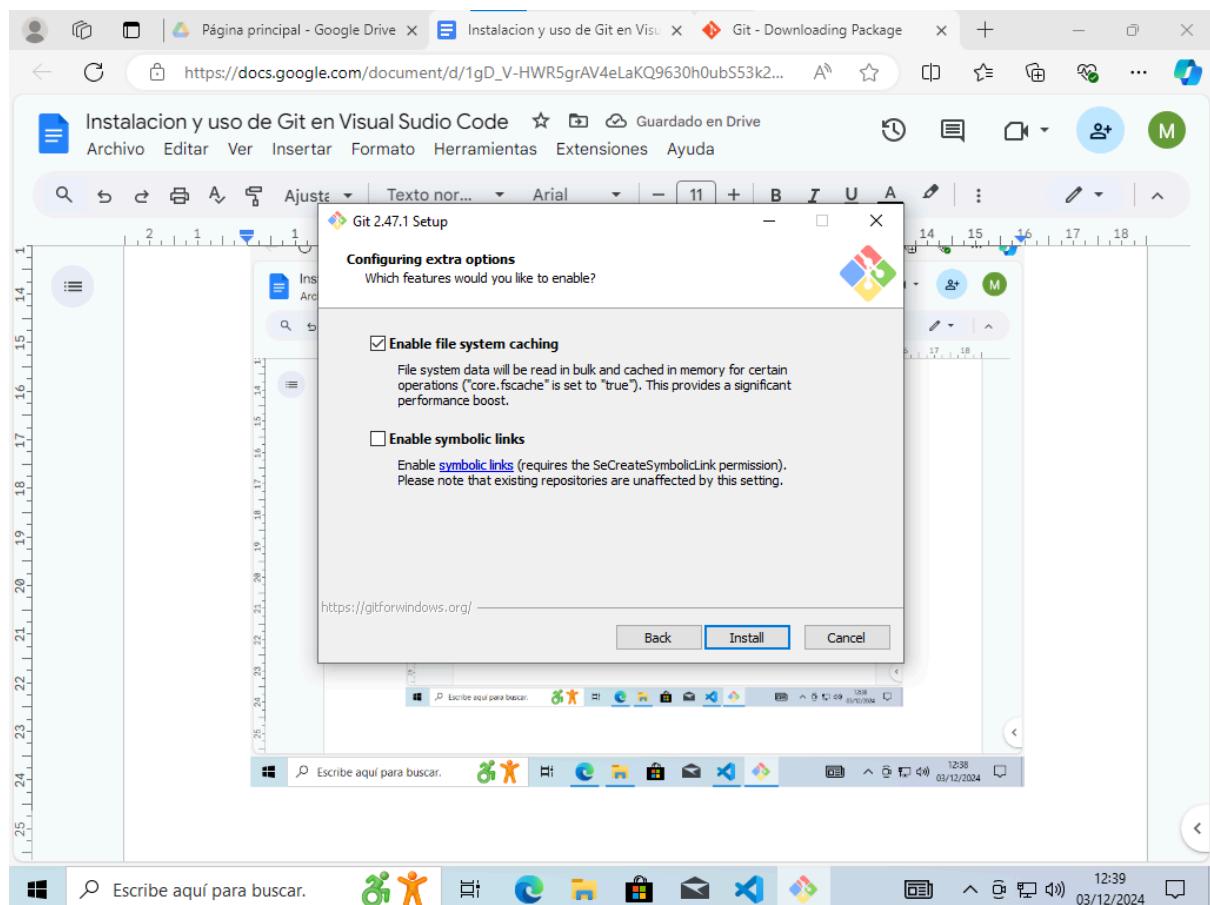




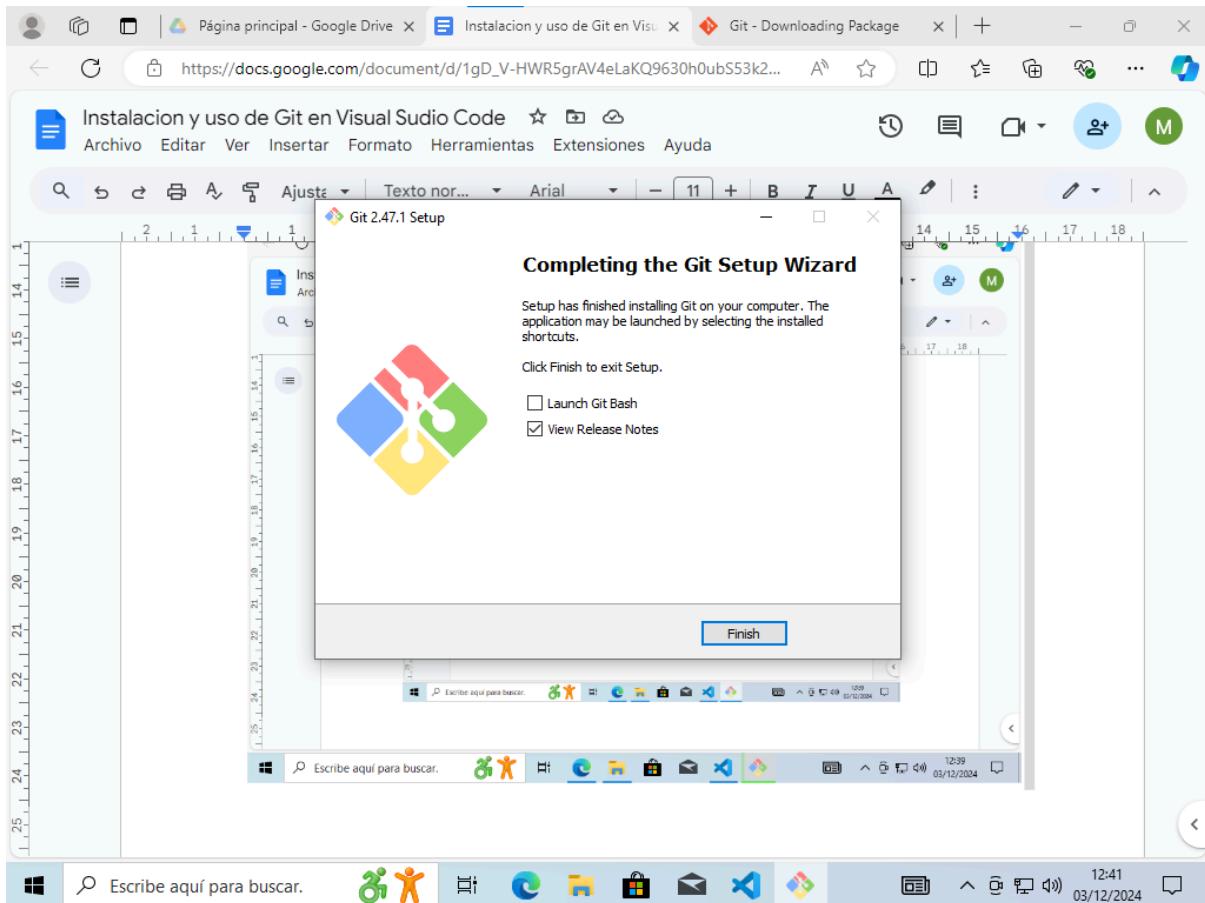




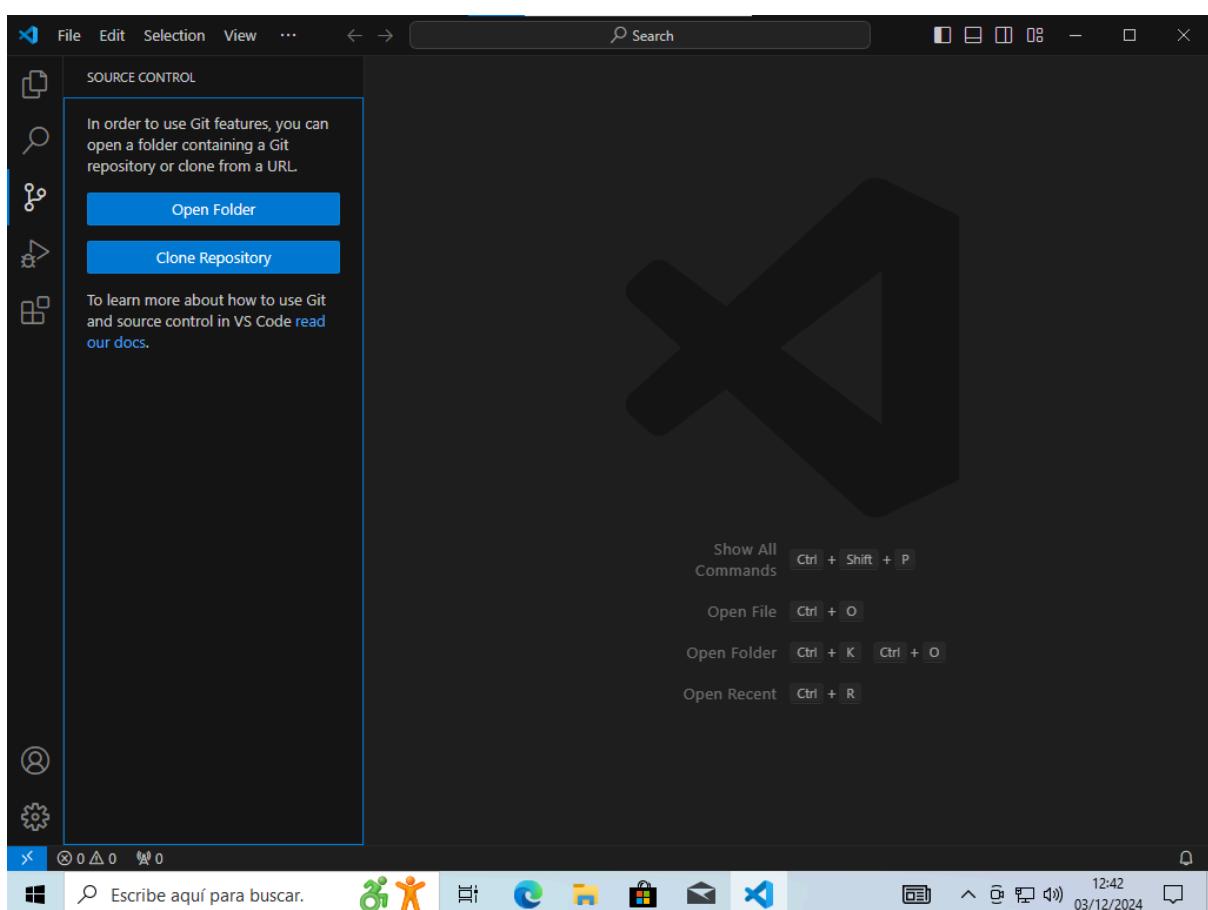
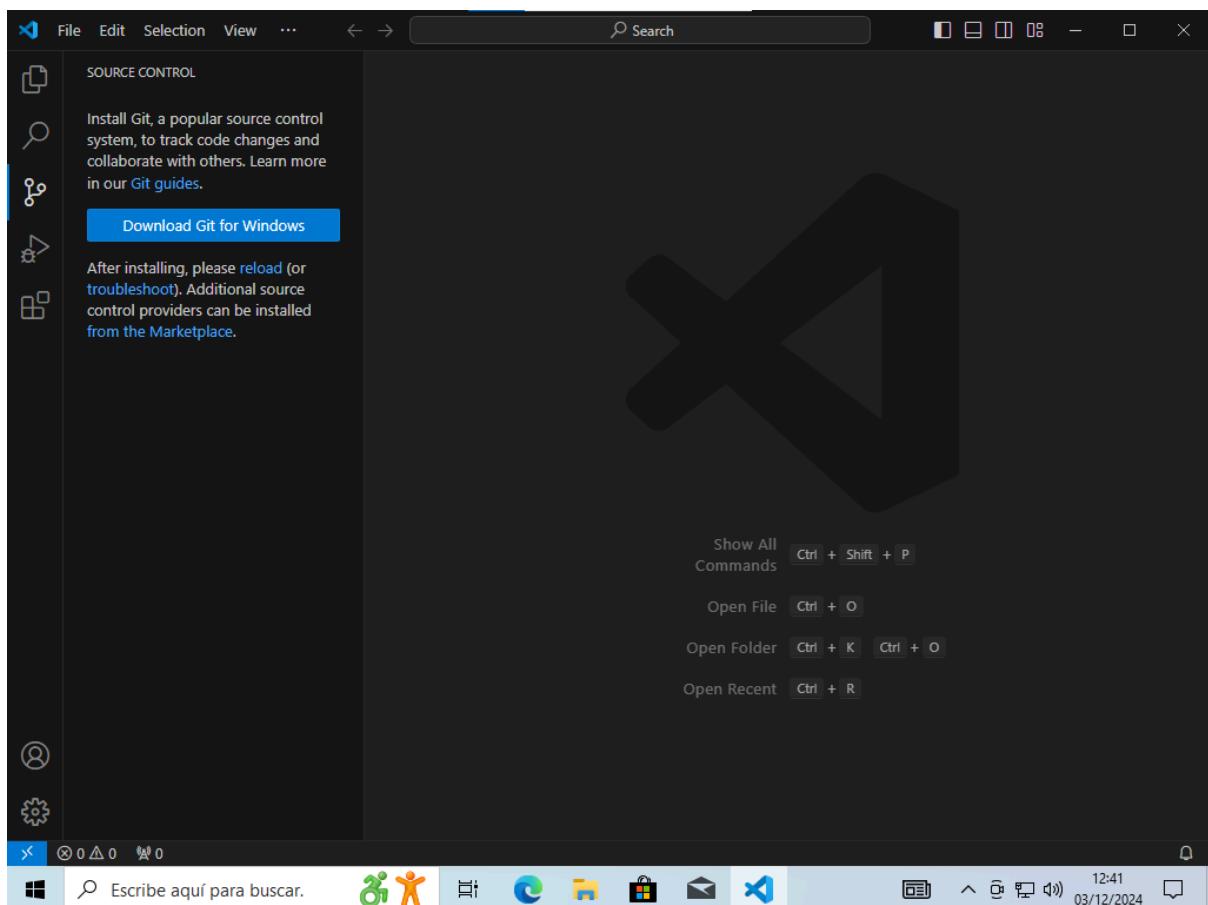




Cuando podamos ver la siguiente ventana significa que nuestro Git está instalado.

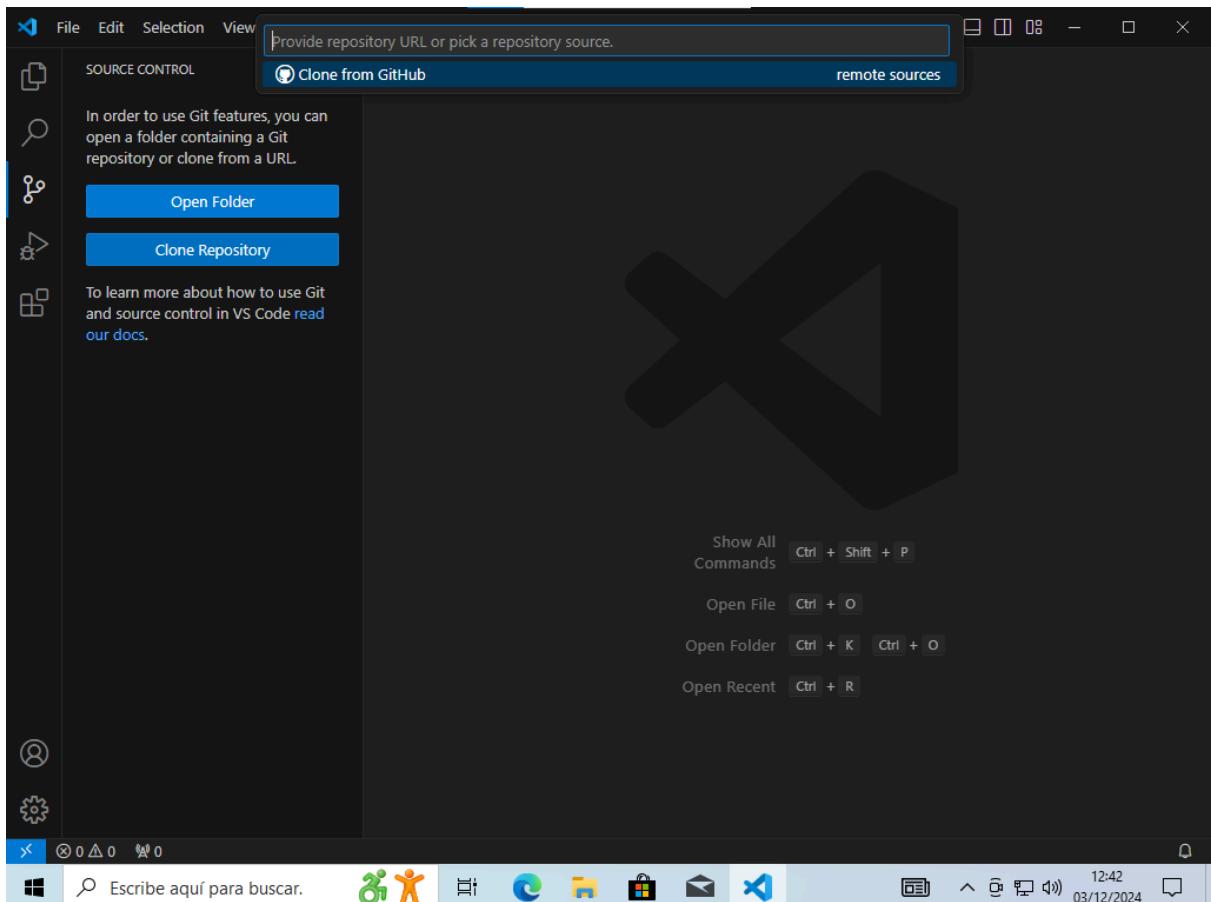


Para finalizar con el proceso de instalación de Git volveremos al entorno de desarrollo a confirmar la instalación. Si pulsamos en el botón azul “reload” o reiniciamos el programa deberán aparecer opciones relacionadas con el control de versiones.

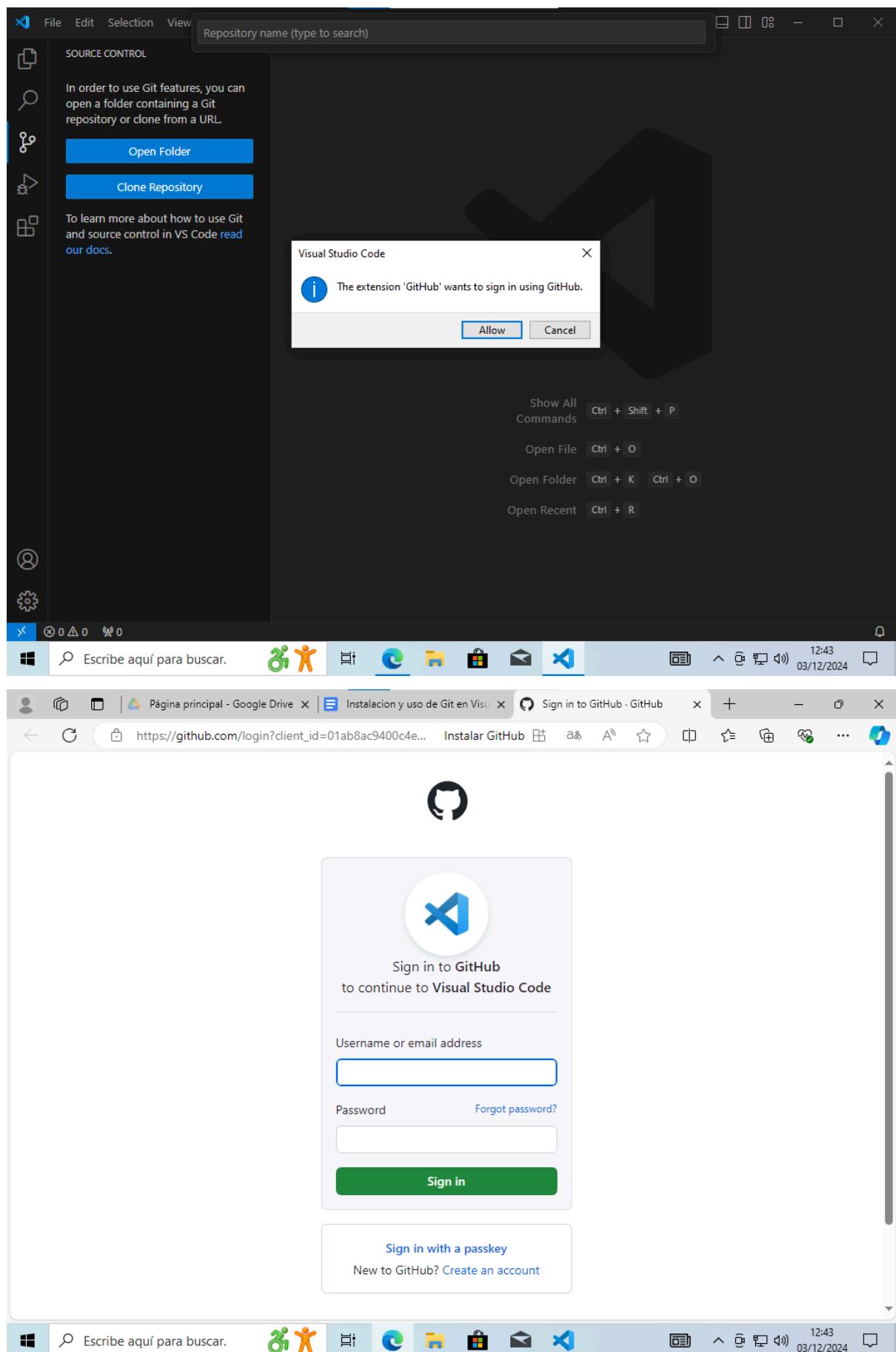


Clonación de un repositorio

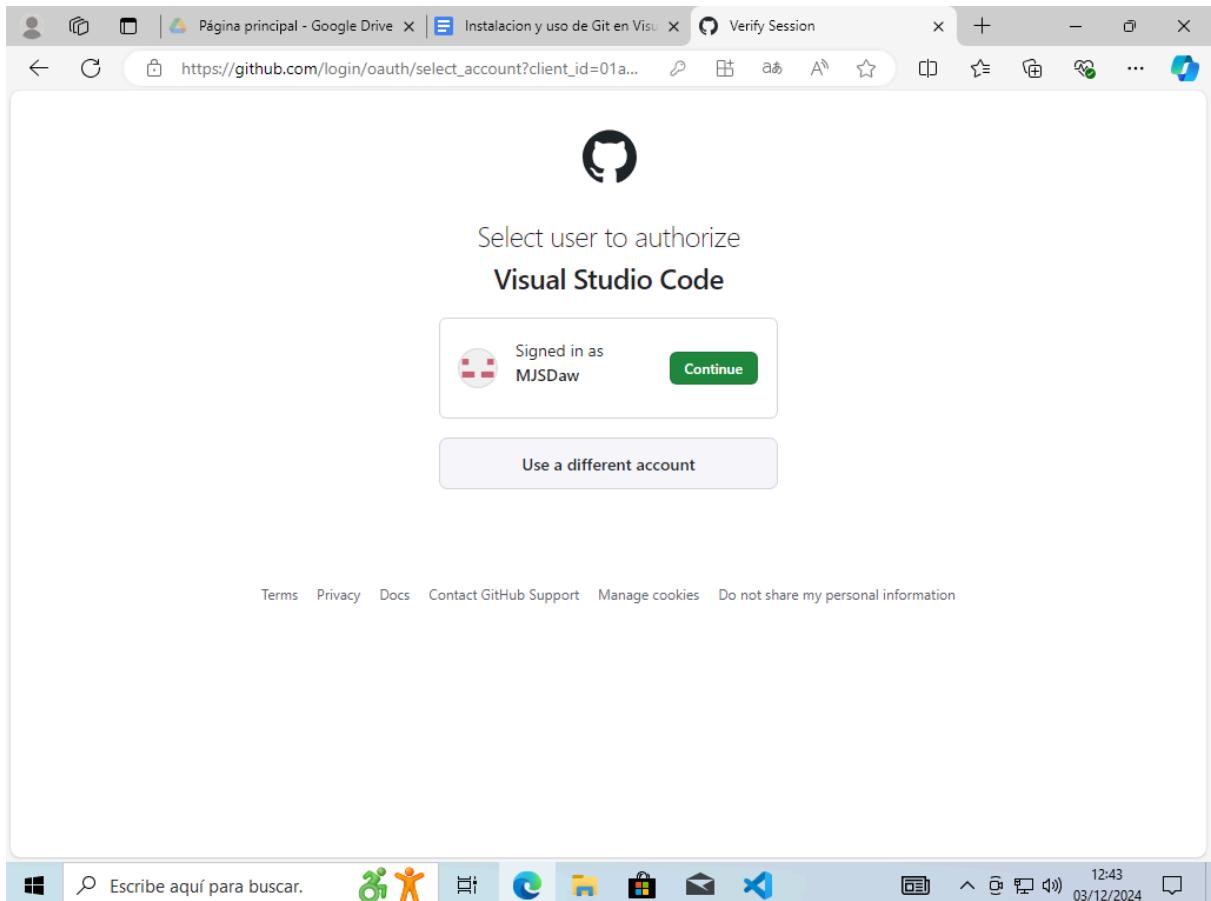
Para clonar un repositorio de GitHub en nuestro equipo debemos seleccionar el botón azul “Clone Repository” / “Clonar Repositorio”. En el panel superior que aparecerá al pulsar seleccionamos la opción “Clone from GitHub” / “Clonar desde GitHub”.



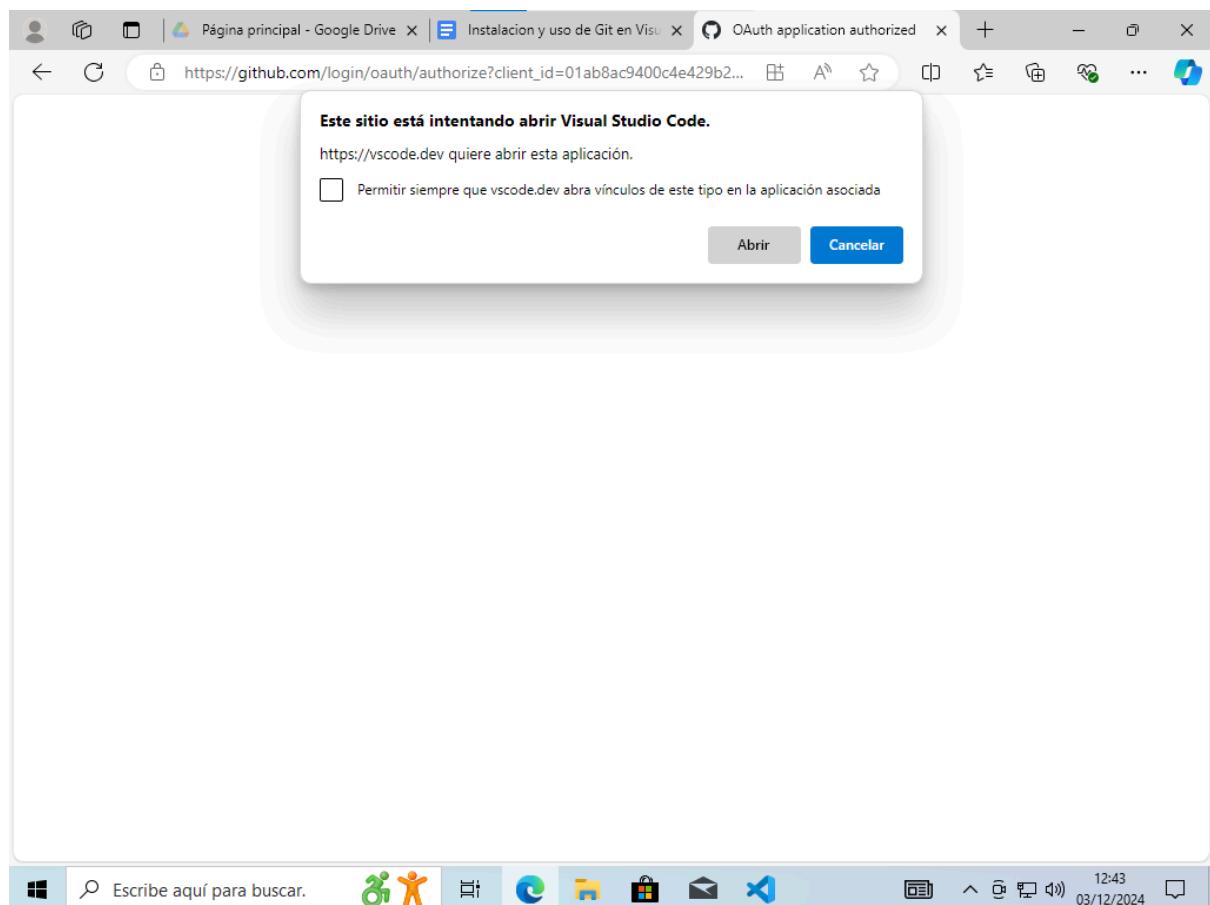
En la ventana emergente pulsaremos en “Allow” / “Permitir”. Esto nos abrirá la página oficial de GitHub para iniciar sesión y enlazar nuestra cuenta con el entorno de desarrollo.



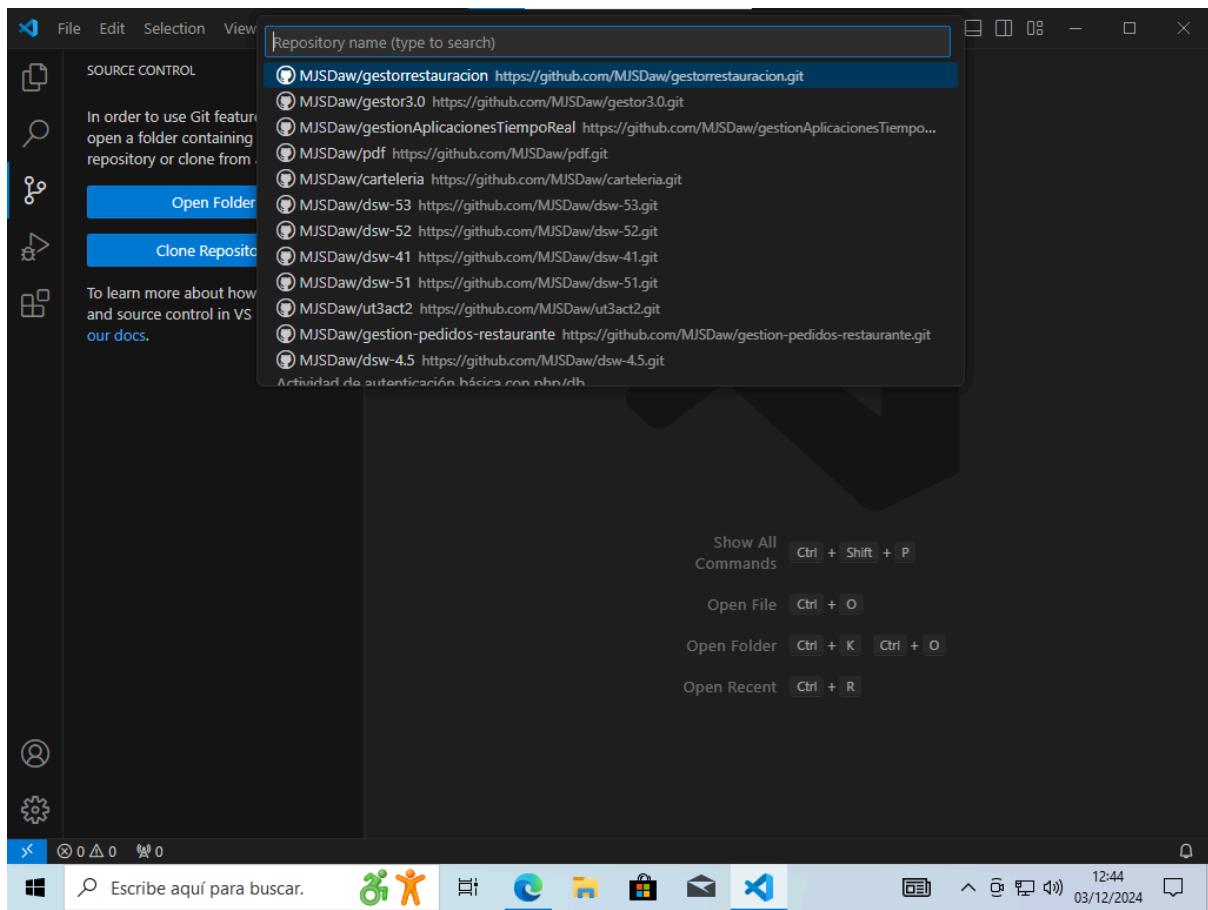
Ingresamos nuestras credenciales para iniciar sesión.



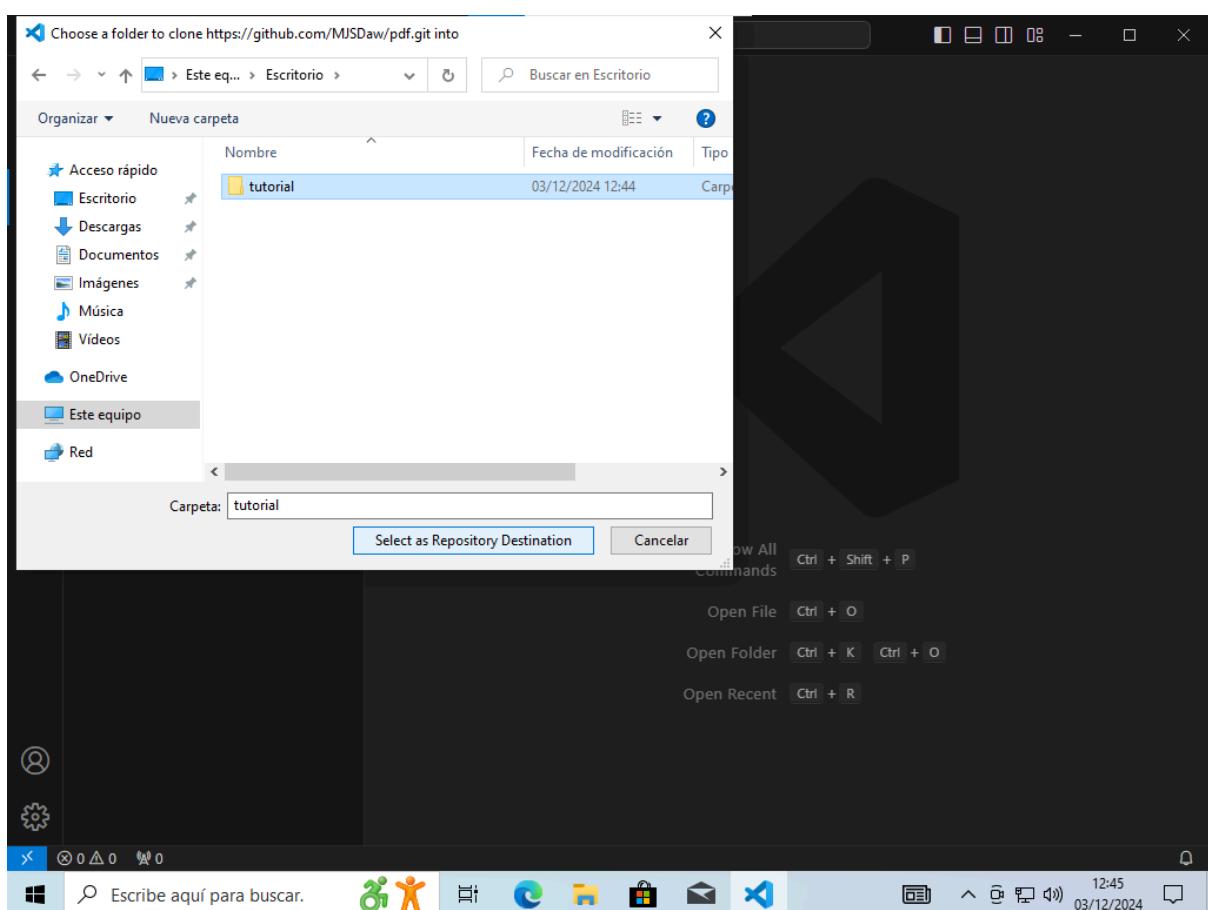
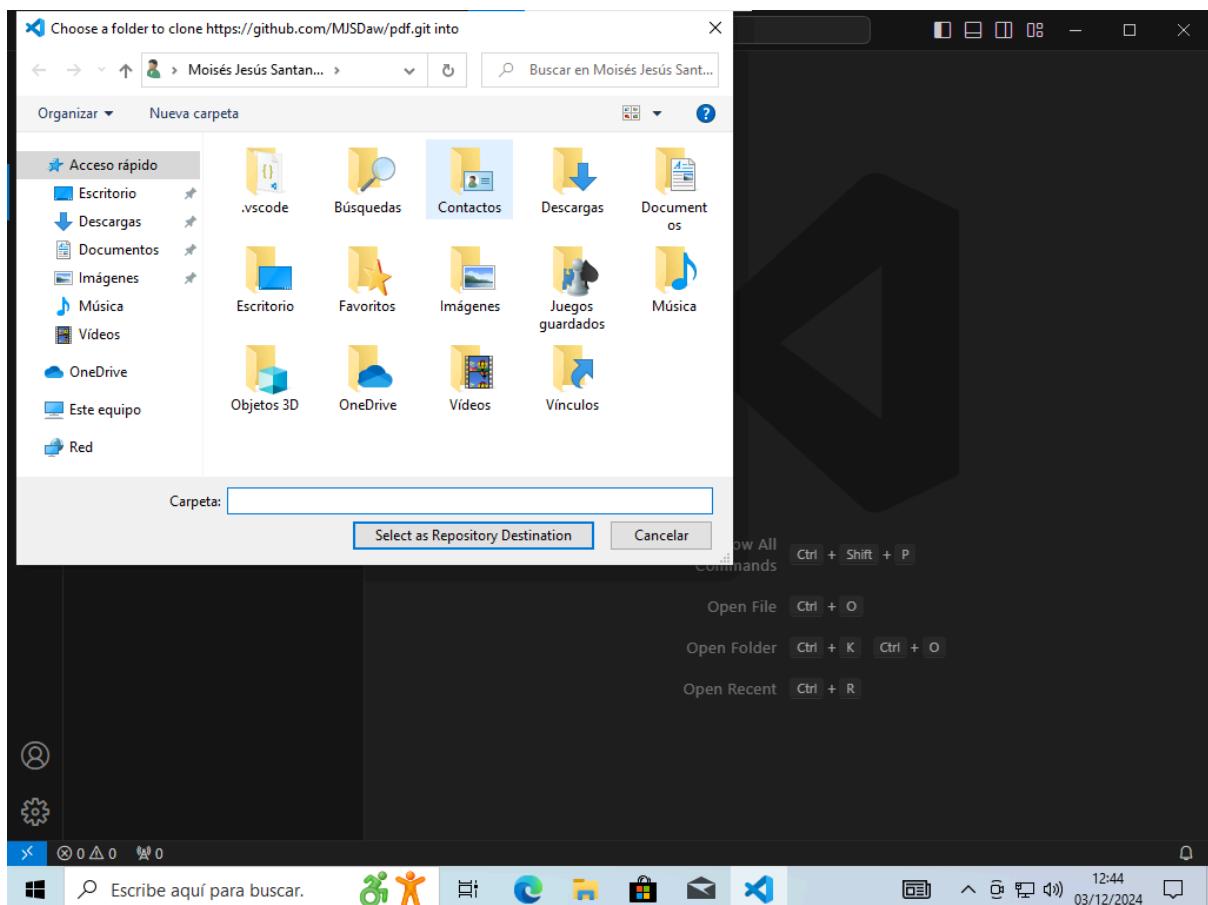
Aceptamos la ventana emergente que nos avisa de que se conectará a Visual Studio Code.

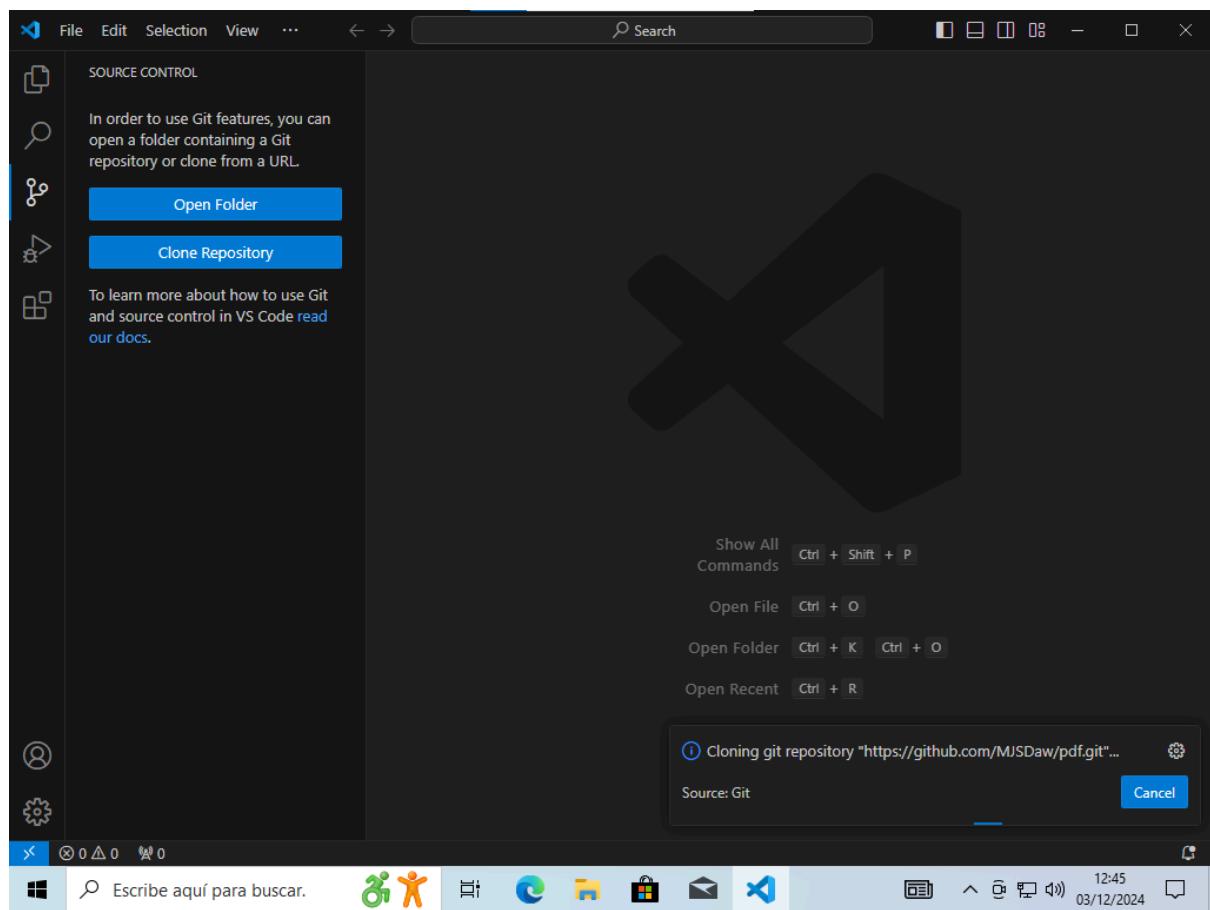


Una vez en el entorno de desarrollo, en el panel superior seleccionaremos el repositorio que queremos clonar.

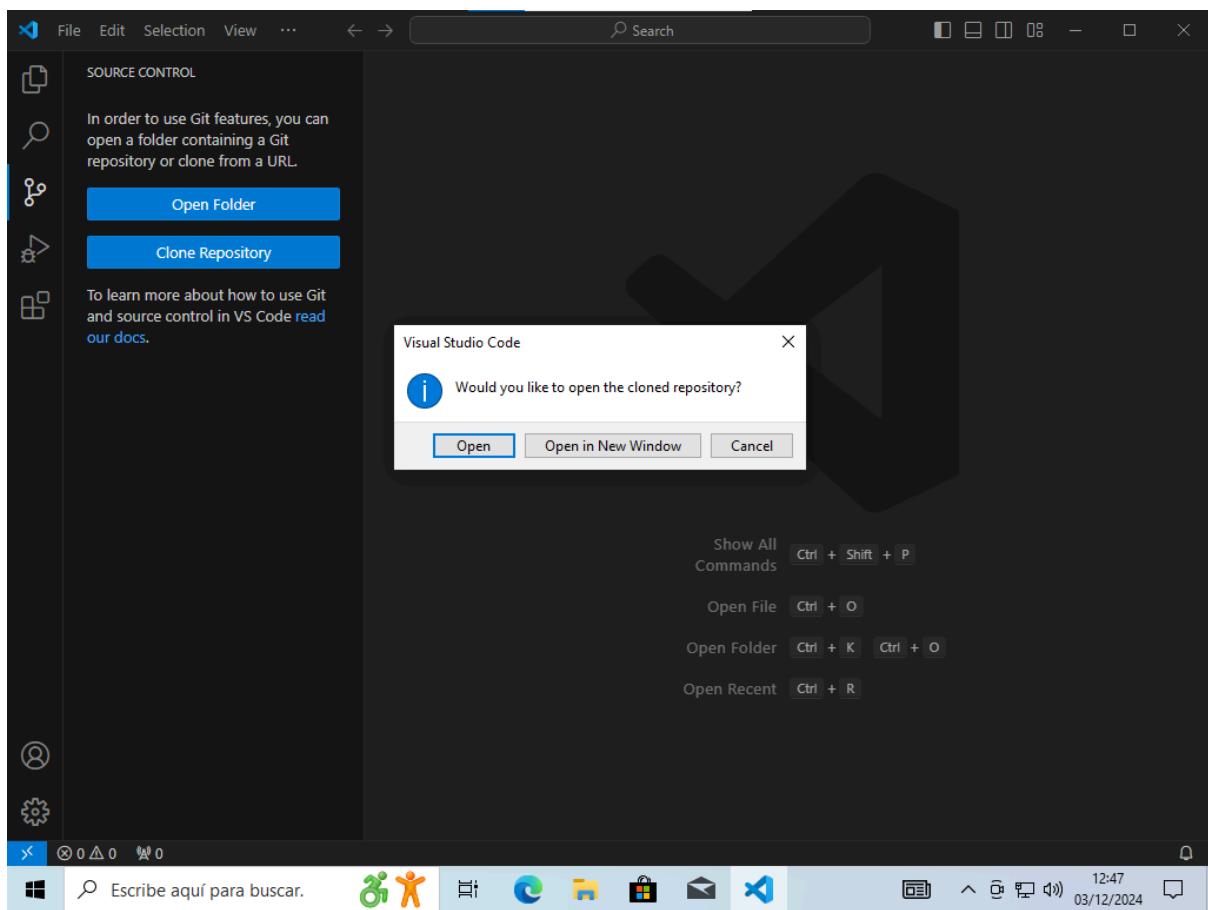


Definimos la ruta donde se clonara nuestro repositorio en el equipo.

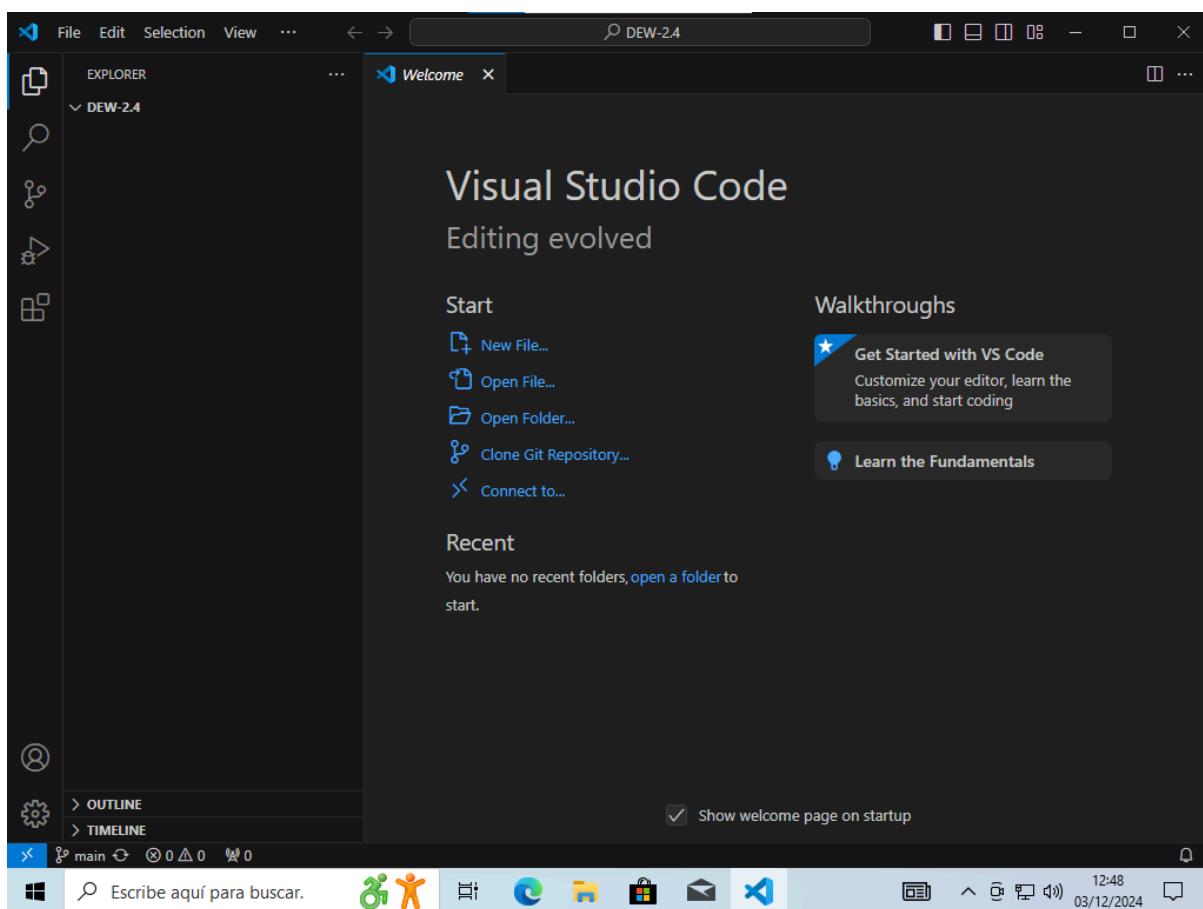
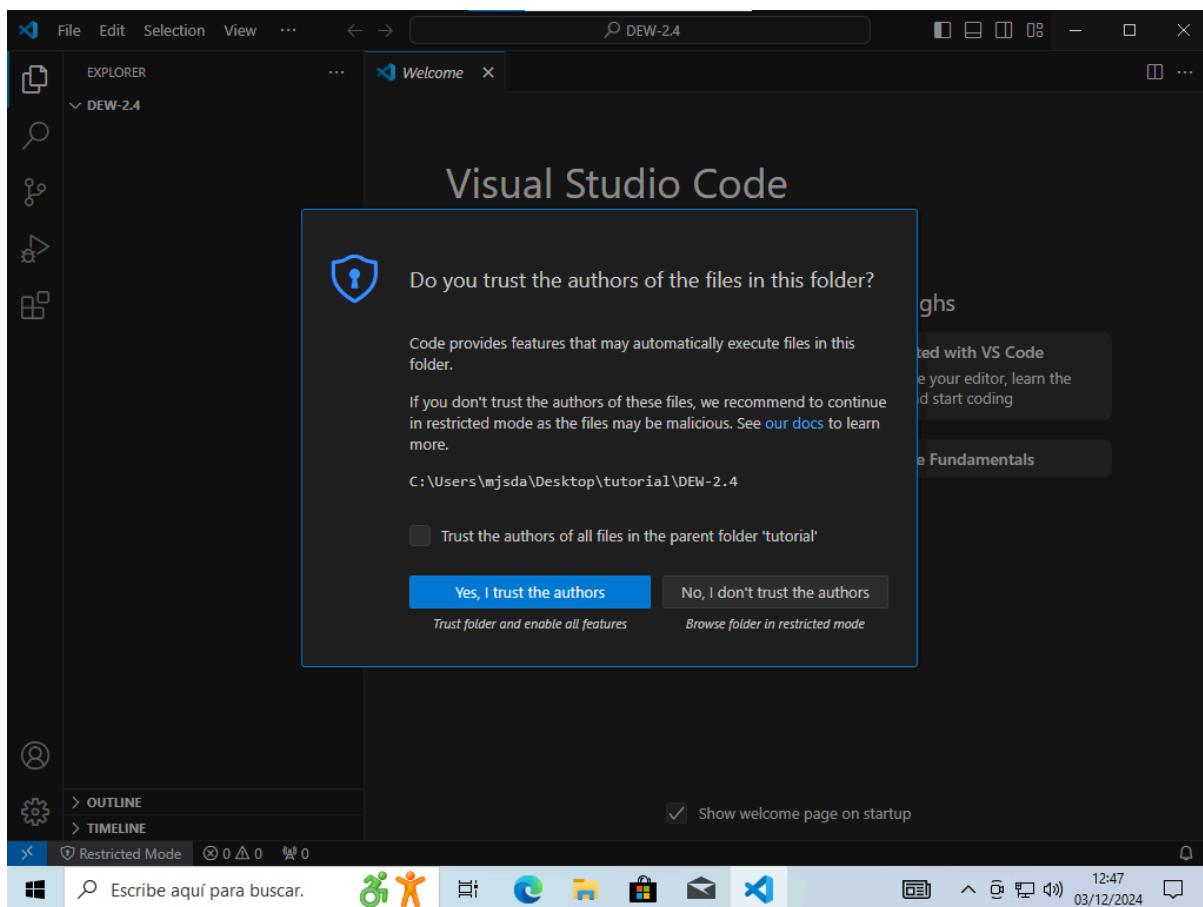




Tras esperar a que se cumpla el proceso decidiremos si abrir nuestro proyecto en el entorno de desarrollo ya abierto ("Open" / "Abrir") o si queremos abrirlo en una nueva pestaña aparte ("Open in New Window" / "Abrir en una nueva ventana").

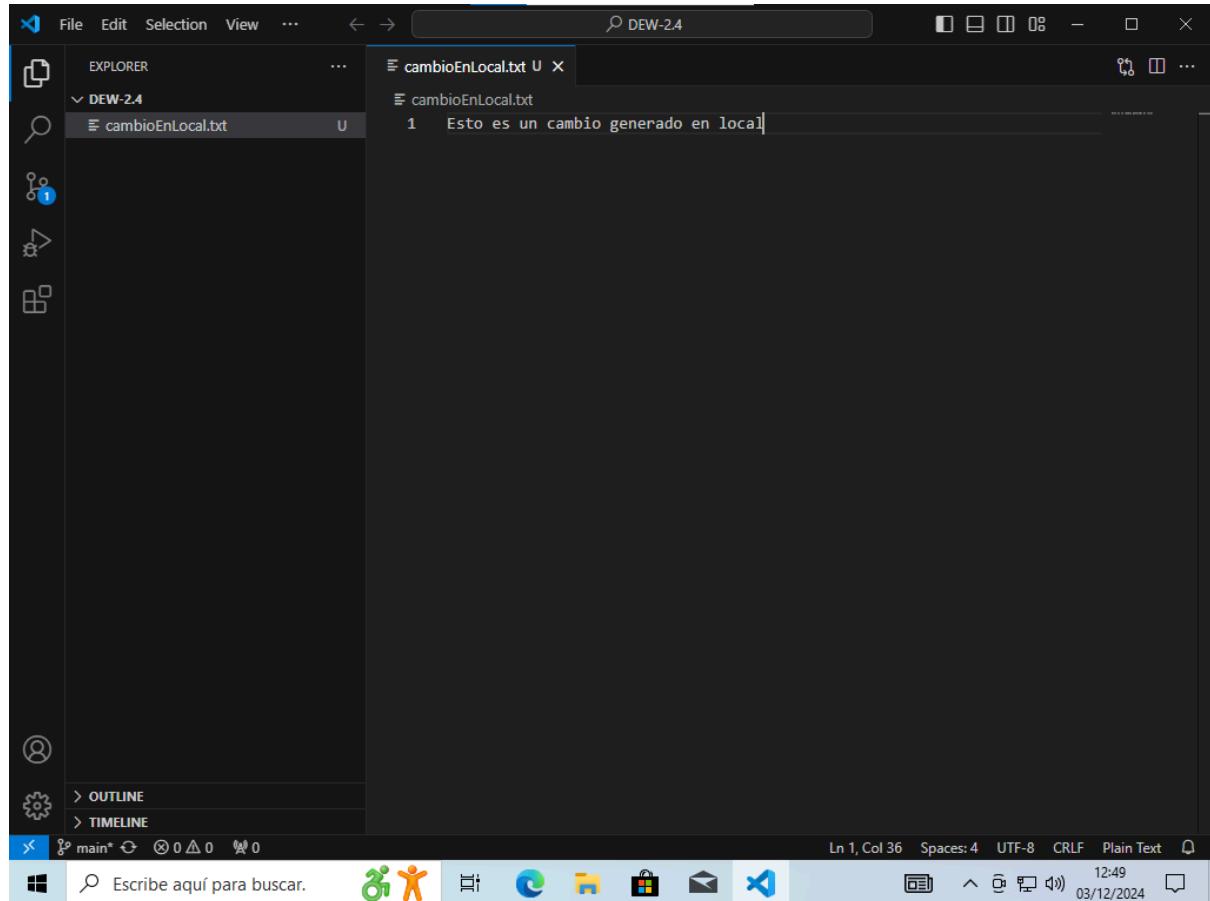


Como último paso Visual Studio Code nos preguntará si confiamos en la proveniencia de dicho proyecto. Pulsamos el botón azul "Yes, I trust the authors" / "Si, confío en los autores".

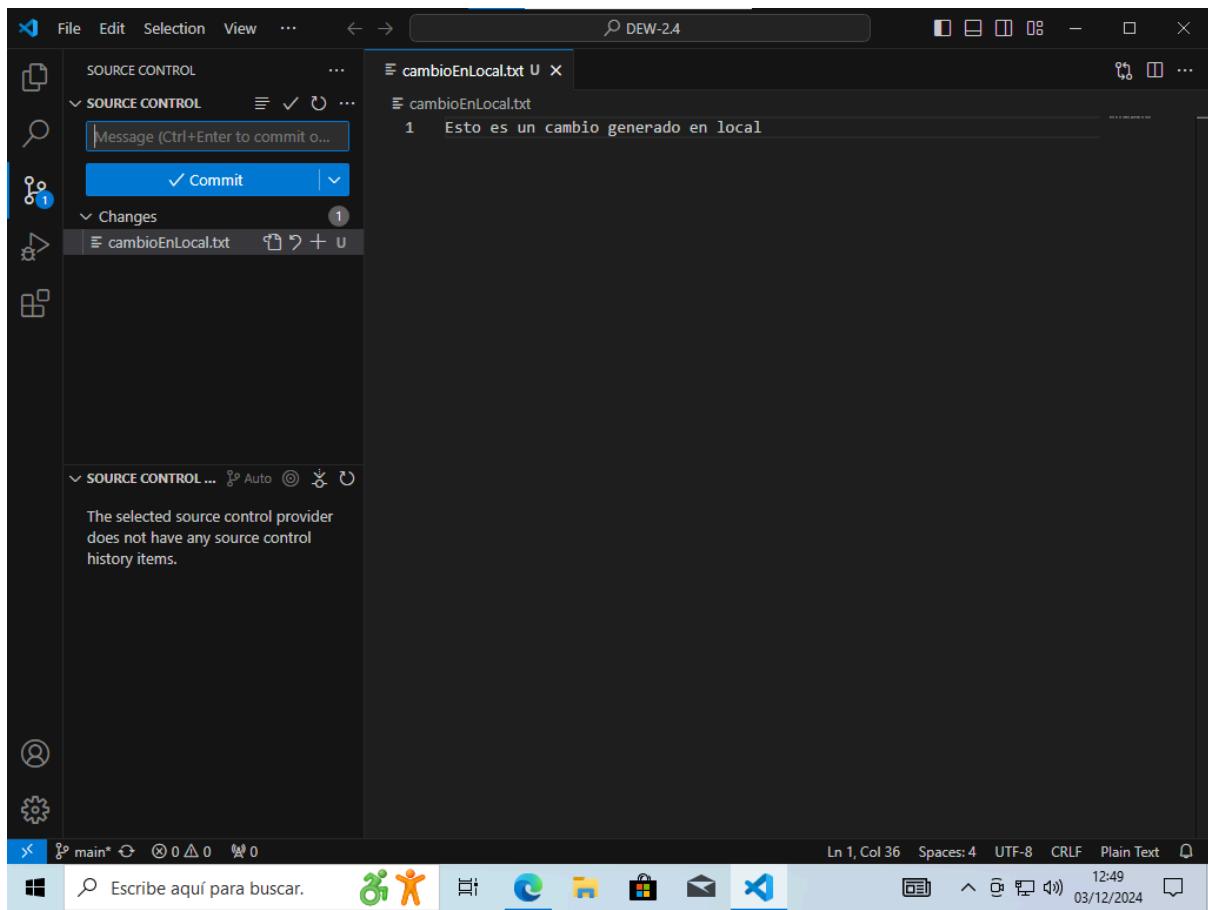


Commit y Push de un proyecto

Para este ejemplo primero debemos generar un archivo, puesto que el repositorio está vacío.

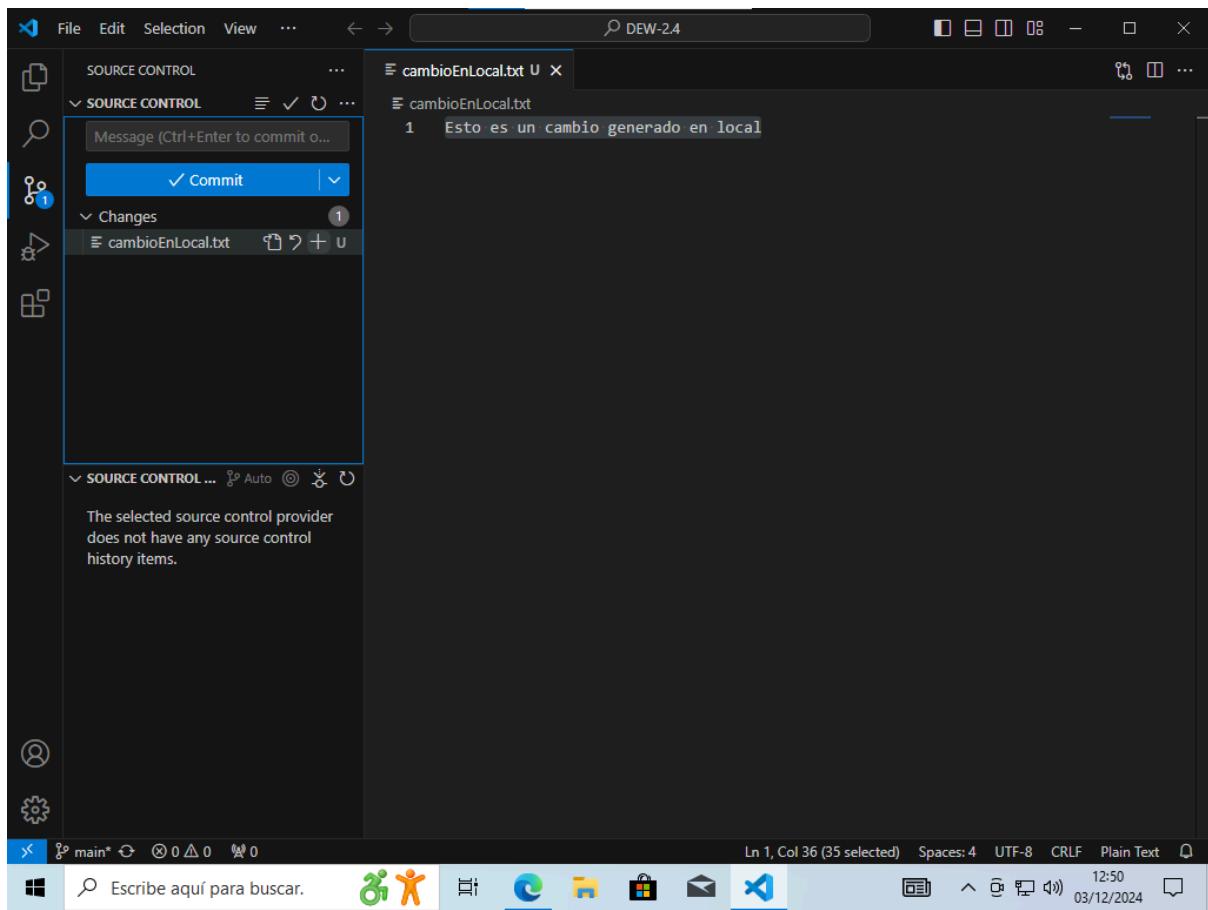


Para realizar un commit debemos dirigirnos al panel de control de versiones de nuestro entorno de desarrollo. Aquí nos mostrará los cambios que se hayan realizado con respecto a cuando clonamos nuestro proyecto (o desde el último commit). Estos cambios los veremos reflejados en la parte de “Changes” / “Cambios”.



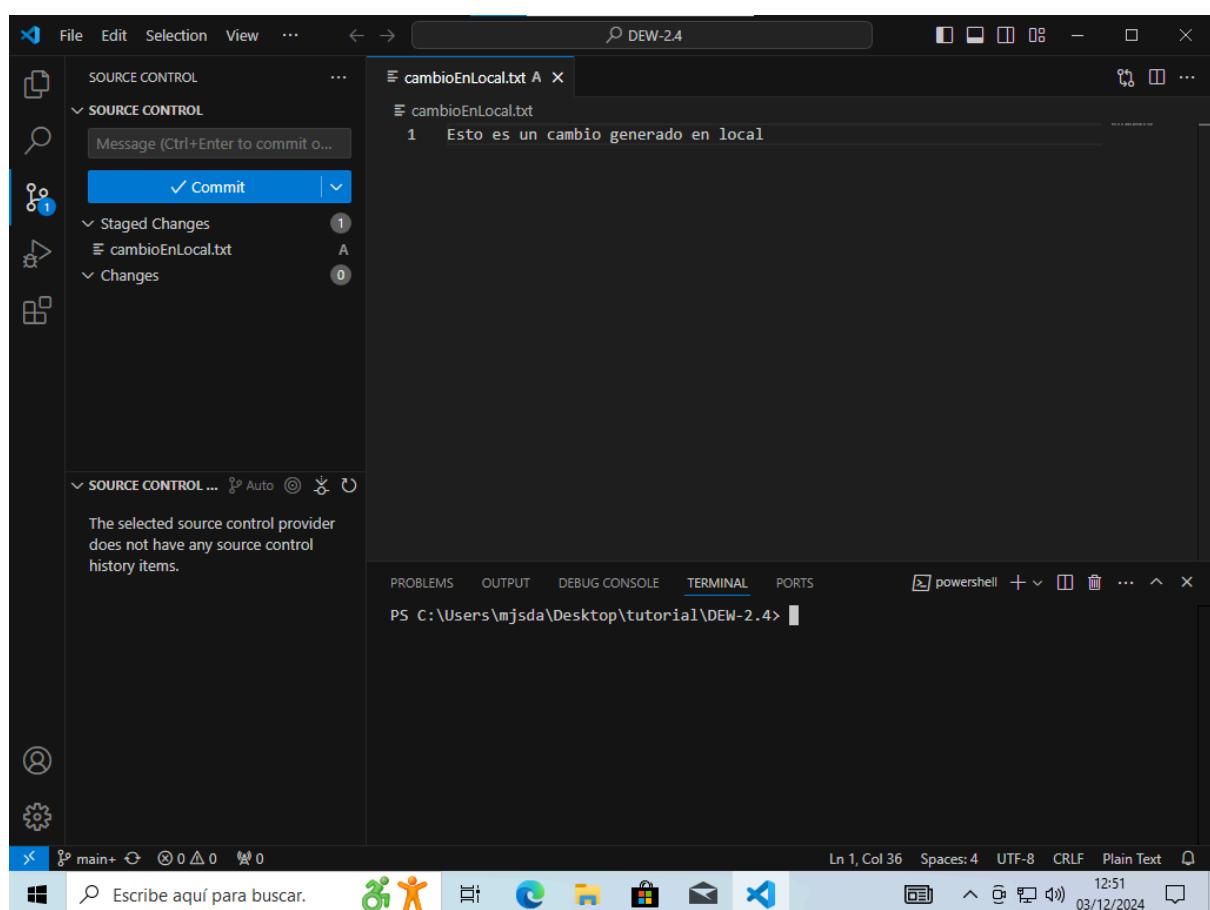
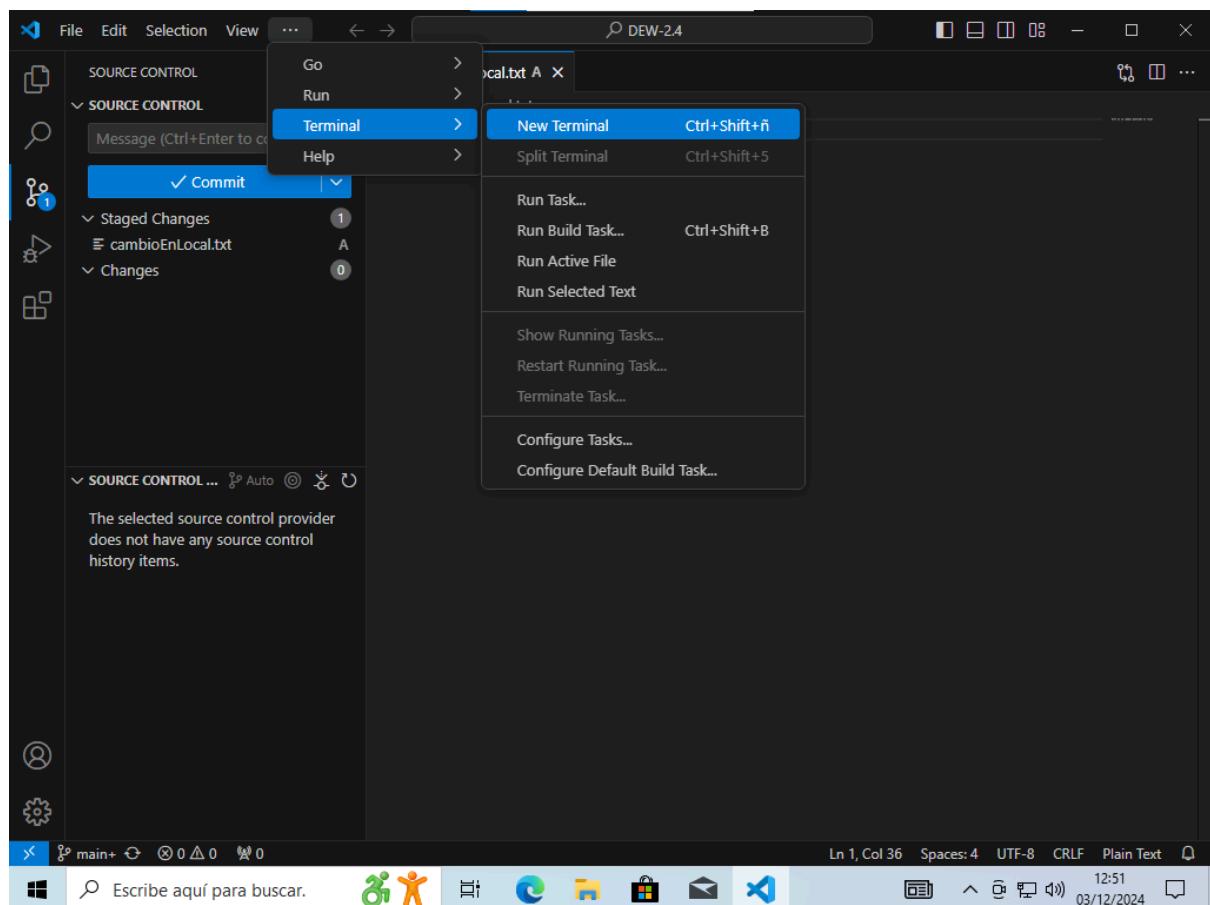
Para poder realizar un commit primero debemos añadir dichos cambios a otra “etapa”, esto lo haremos pulsando en el símbolo “+” que aparece a la derecha de “Changes”.

Para agilizar este paso, si pulsamos directamente “Commit” sin cambiar de fase los cambios nos saldrá una ventana emergente que nos avisa de que hará este cambio de fase automáticamente. Si te interesa esa opción, en la misma ventana emergente puedes pulsar la opción para que lo haga siempre que le pulses “Commit”.



En un proceso normal ahora simplemente pulsamos “Commit”. Pero al ser la primera vez que usamos GitHub con nuestro Visual Studio Code, primero debemos registrar nuestro usuario y correo.

Para ello abriremos una terminal desde las opciones del borde superior como se muestra en la imagen siguiente.



En la consola pondremos las 2 siguientes lineas que puedes copiar y pegar, sustituyendo las letras en naranja por tu correo y tu usuario en ese orden.

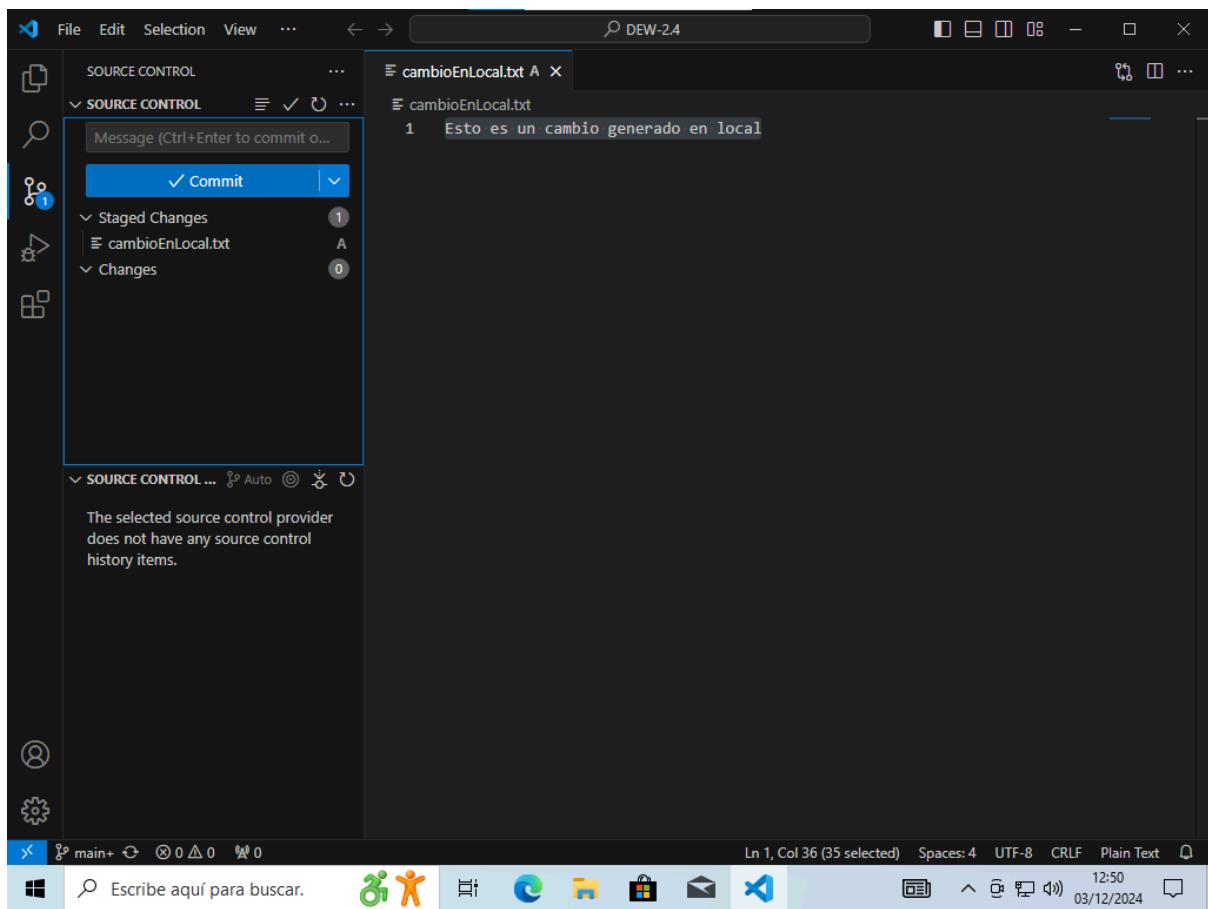
```
git config --global user.email "you@example.com"  
git config --global user.name "Your Name"
```

Tras sustituir con nuestra información, deberíamos poder ejecutarlo sin problemas.

The screenshot shows a dark-themed IDE interface. On the left, there's a sidebar with icons for file operations, source control, search, and settings. The main area has a title bar "DEW-2.4". In the center-left, a "SOURCE CONTROL" panel is open, showing a "Message (Ctrl+Ente...)" field containing "cambioEnLocal.txt A" and a "Commit" button. Below it, "Staged Changes" show "cambioEnLocal.txt A" and "Changes" show "0". A message in the status bar says "The selected source control provider does not have any source control history items." To the right of the source control panel is a code editor with the file "cambioEnLocal.txt" open, displaying the text "1 Esto es un cambio generado en local". At the bottom of the code editor is a status bar with "Ln 1, Col 36 Spaces: 4 UTF-8 CRLF Plain Text". Below the code editor is a terminal window titled "powershell" showing the command: PS C:\Users\mjdsda\Desktop\tutorial\DEW-2.4> git config --global user.email "mjdsdaw@gmail.com" >> git config --global user.name "MJSDAW". The terminal status bar shows "12:52 03/12/2024".

This screenshot is nearly identical to the one above, showing the same IDE interface, source control panel, code editor, and terminal window. The only difference is in the terminal window, where the command "git config --global user.name "MJSDAW"" has been executed, resulting in the output "PS C:\Users\mjdsda\Desktop\tutorial\DEW-2.4> git config --global user.name "MJSDAW"".

Ahora podemos realizar el commit sin problema ninguna pulsando “Commit”.



Añadiremos un mensaje describiendo el commit realizado como se muestra en el ejemplo, guardaremos y pulsaremos el “tick” que se ubica en la parte superior a la derecha.

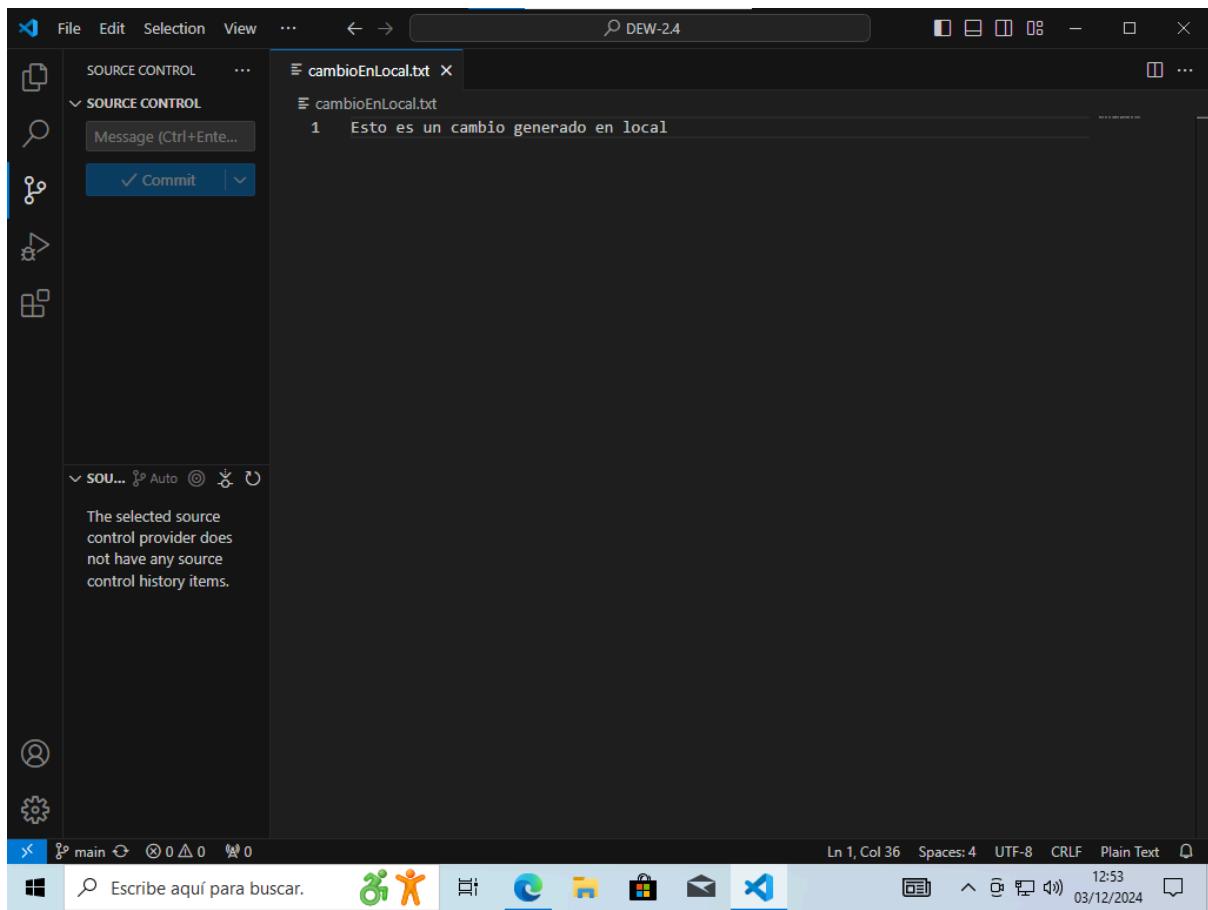
The screenshot shows a dark-themed code editor interface. On the left, there's a sidebar with icons for file operations like Open, Save, and Undo/Redo, along with a Source Control section. The Source Control section has a dropdown menu set to 'Message (Ctrl+Enter...)'. Below it are buttons for 'Commit' (with a checkmark icon) and 'Staged Changes' (with a 1 icon). A 'Changes' section shows 0 items. The main area displays a file named 'cambioEnLocal.txt' with a status 'A'. The file content is a Git commit message template:

```
git > COMMIT_EDITMSG
1 |
2 # Please enter the commit message for your changes. Lines starting
3 # with '#' will be ignored, and an empty message aborts the commit.
4 #
5 # On branch main
6 #
7 # Initial commit
8 #
9 # Changes to be committed:
10 #   new file: cambioEnLocal.txt
11 #
12
```

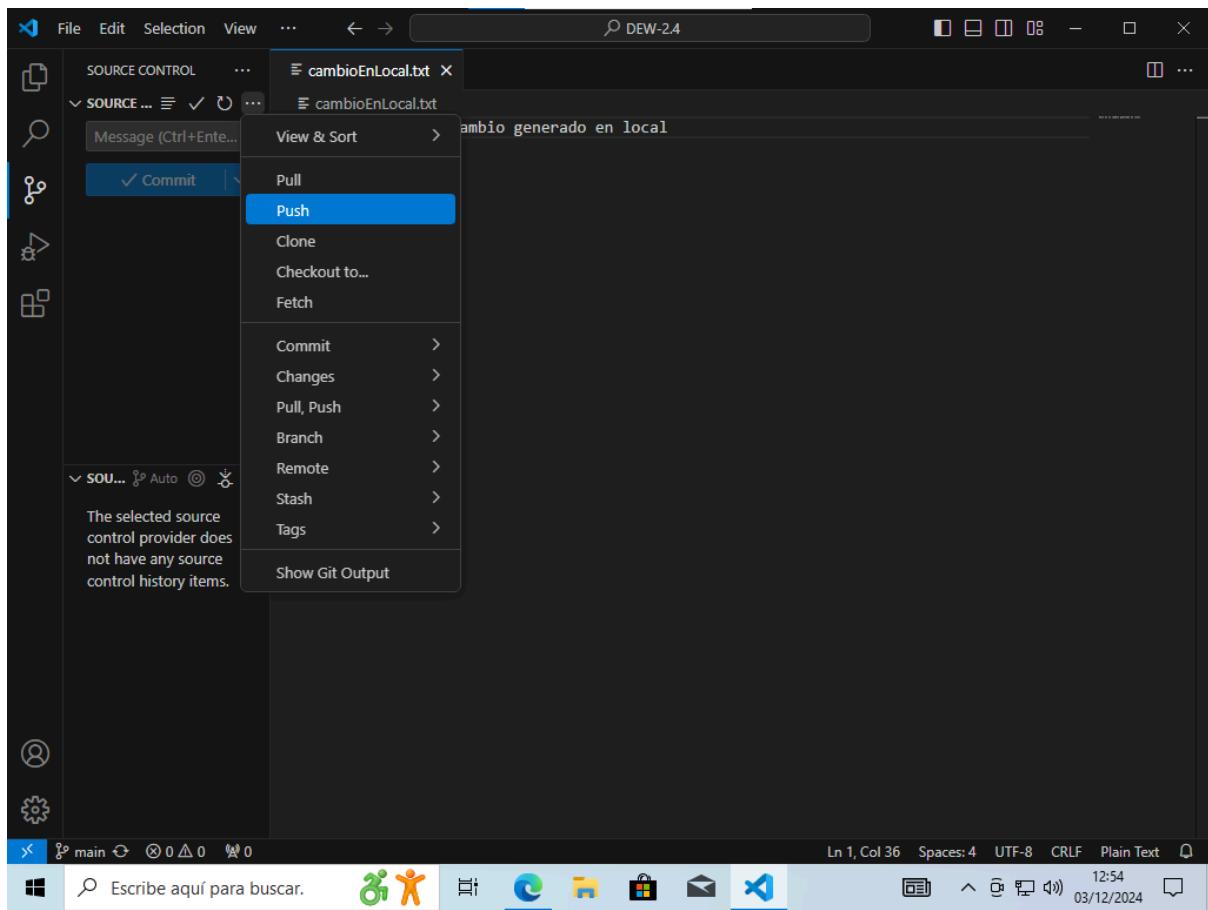
Below the main area, a tooltip says: 'The selected source control provider does not have any source control history items.' The bottom status bar shows 'Ln 1, Col 1' and 'Spaces: 4'.

This screenshot shows the same code editor interface after a commit message has been entered. The commit message 'Realizado un cambio en el local.' has been added to the template. The bottom status bar now shows 'Ln 1, Col 33'.

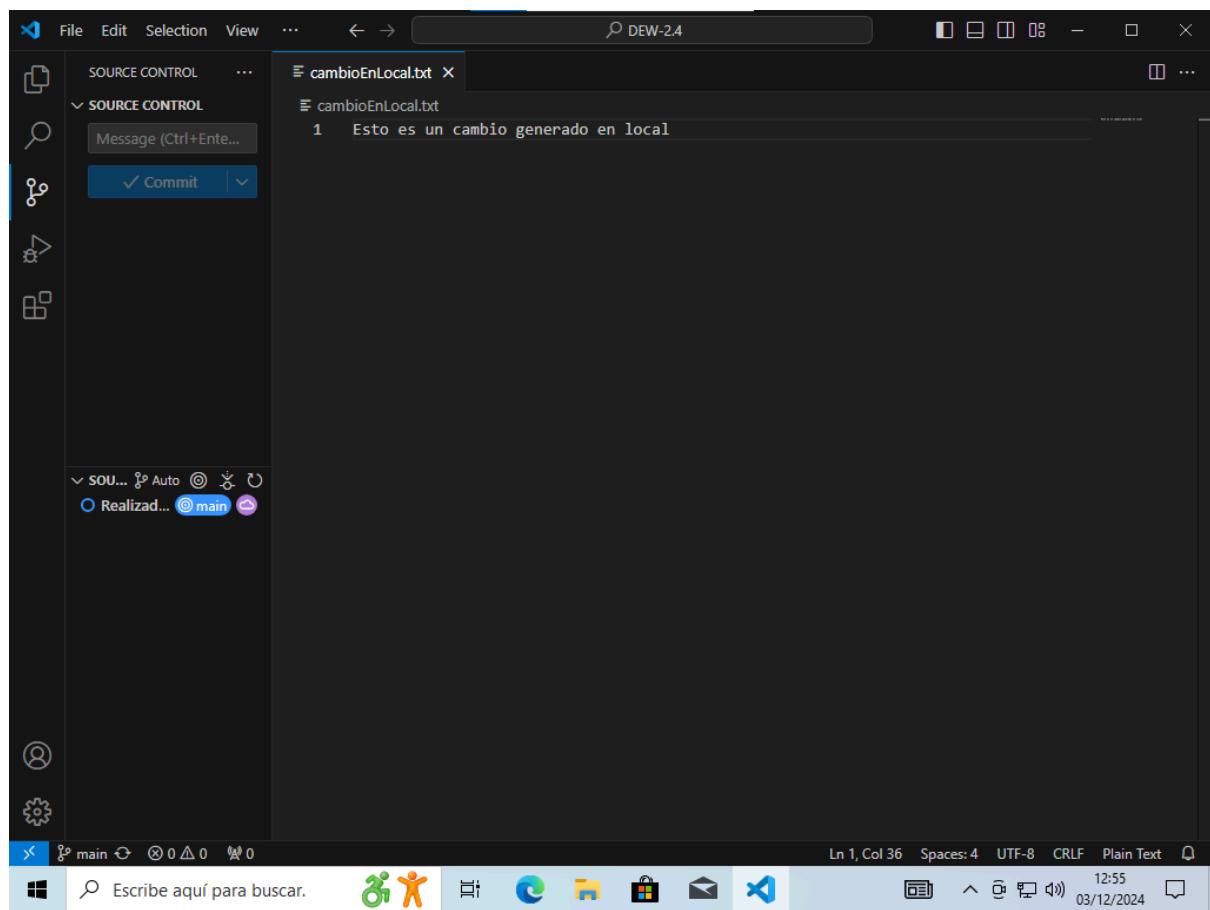
```
git > COMMIT_EDITMSG
1 Realizado un cambio en el local.
2
3 # Please enter the commit message for your changes. Lines starting
4 # with '#' will be ignored, and an empty message aborts the commit.
5 #
6 # On branch main
7 #
8 # Initial commit
9 #
10 # Changes to be committed:
11 #   new file: cambioEnLocal.txt
12 #
```



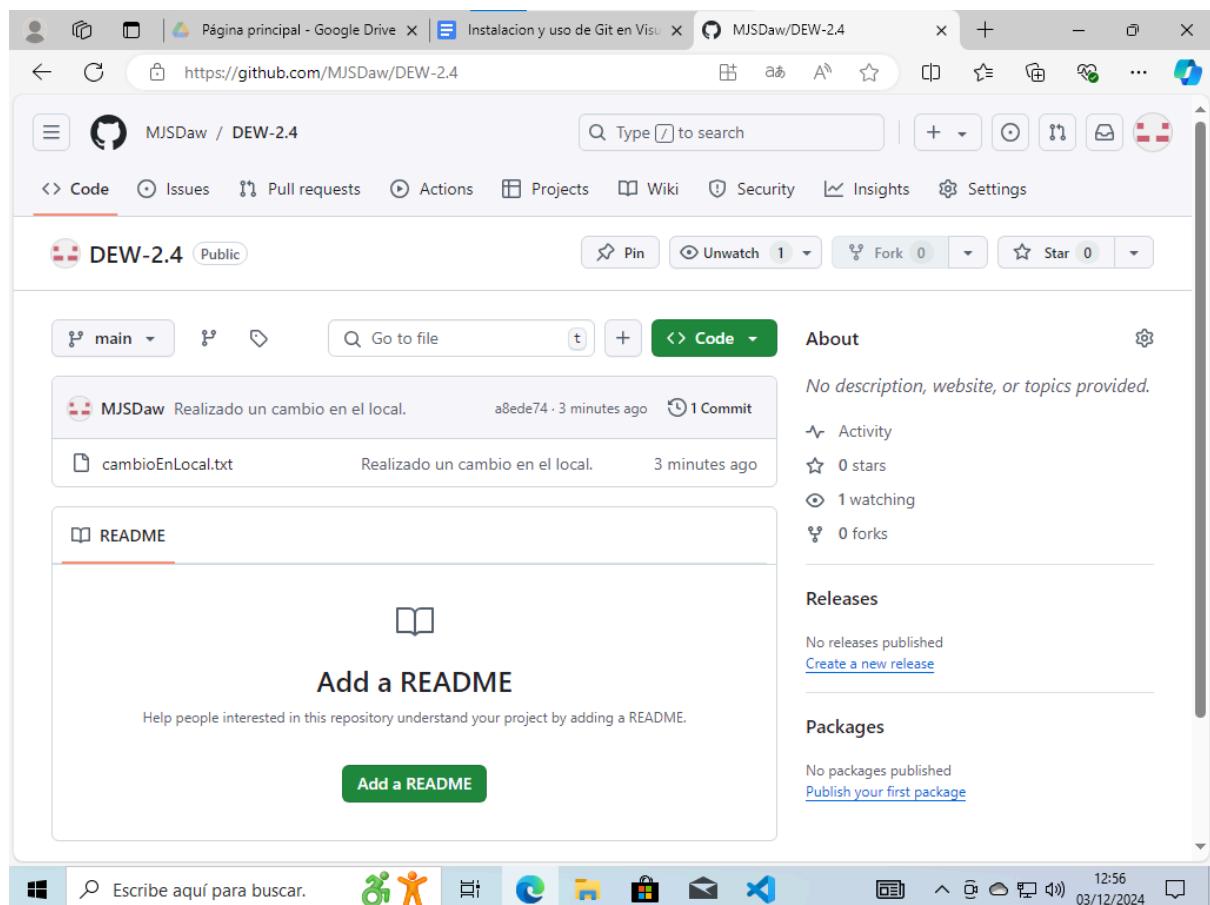
Una vez concluido el commit solo nos queda hacer un push para subirlo a GitHub, esto lo logramos pulsando los 3 puntos que se destacan en la imagen siguiente del panel lateral, se nos desplegará un menú donde pulsaremos "Push".



Tras hacer el push veremos en la parte inferior del panel lateral como se nos crea un “checkpoint” en nuestro historial de versiones. En ese panel se registran todos los cambios de nuestro proyecto.



Si accedemos a nuestro repositorio a través de la página oficial de GitHub, podemos confirmar el cambio realizado.



Pull

Primero debe haber una actualización en GitHub para poder hacer un pull a nuestro equipo local.

MJSDaw / DEW-2.4

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main / DEW-2.4 /

MJSDaw Create cambioEnGitHub.txt 7091292 · now History

Name	Last commit message	Last commit date
cambioEnGitHub.txt	Create cambioEnGitHub.txt	now
cambioEnLocal.txt	Realizado un cambio en el local.	4 minutes ago

File Edit Selection View ... DEW-2.4

EXPLORER

DEW-2.4 cambioEnLocal.txt

Show All Commands Ctrl + Shift + P

Go to File Ctrl + P

Find in Files Ctrl + Shift + F

Toggle Full Screen F11

Show Settings Ctrl + ,

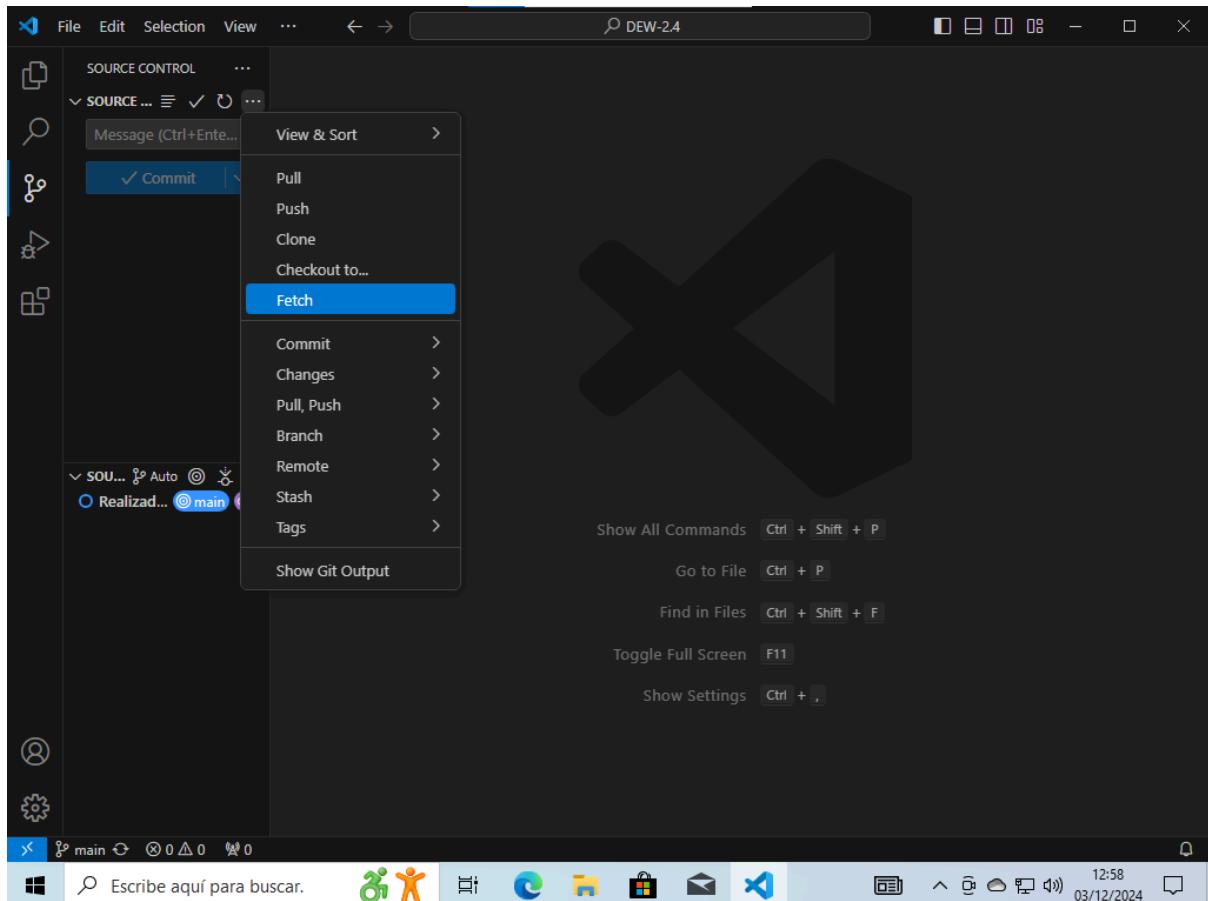
OUTLINE

TIMELINE

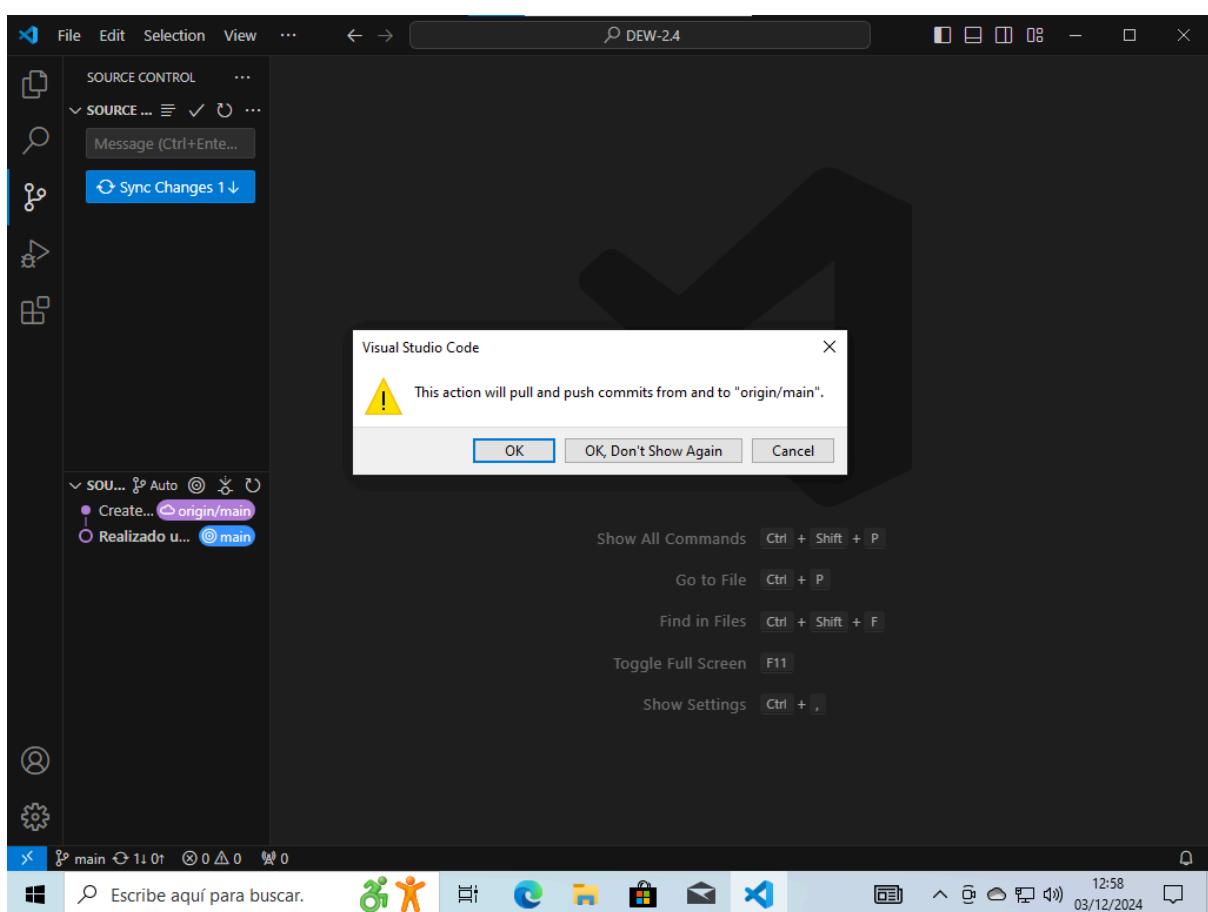
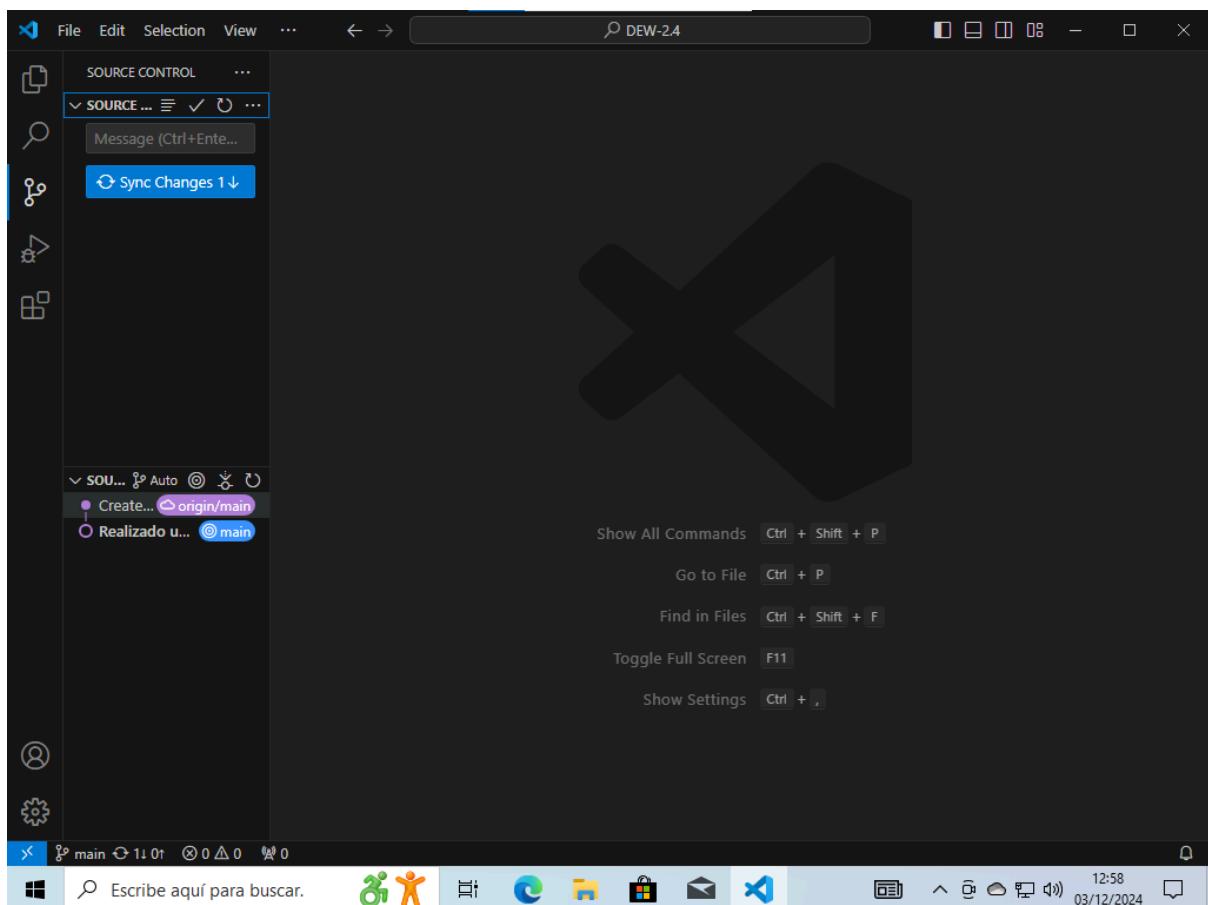
main

Escribe aquí para buscar.

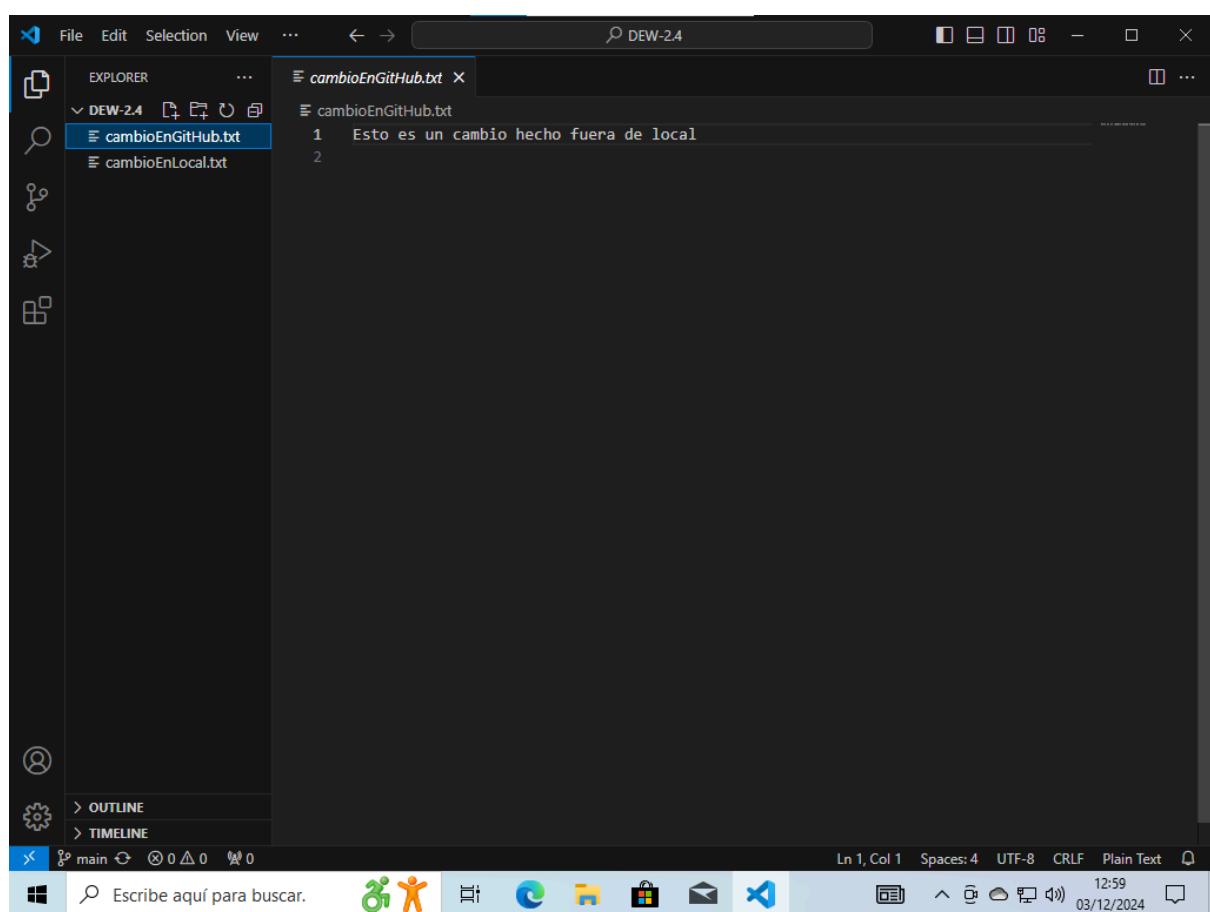
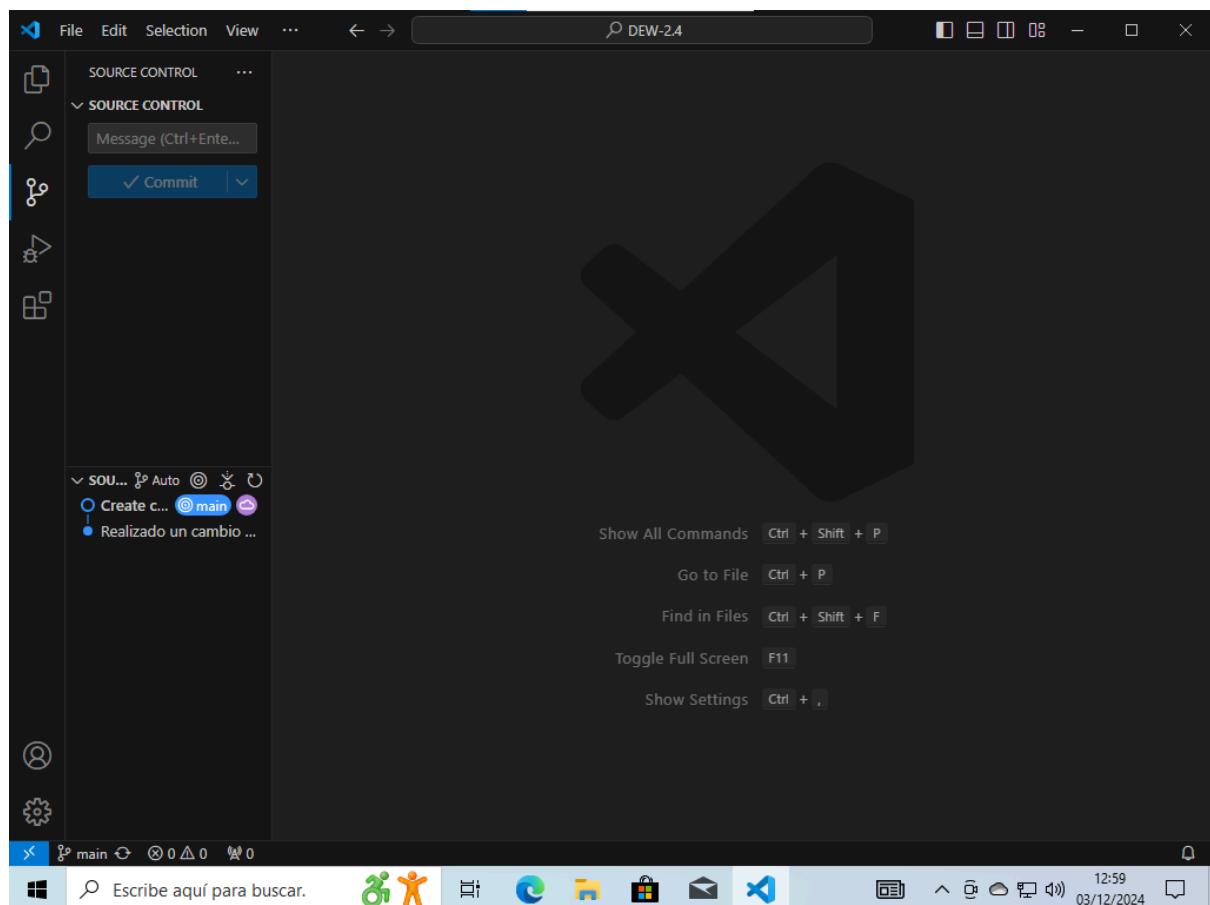
Nos dirigimos al apartado de control de versiones y en el menú de opciones pulsaremos “Fetch” con el objetivo de confirmar que hay cambios.



En nuestra ventana lateral inferior podemos ver como se nos muestra que hay una versión en GitHub posterior a la de nuestro equipo. Para hacer pull podemos hacerlo pulsando “Pull” desde el mismo menú contextual o pulsando “Sync Changes” que nos aparece tras hacer el fetch.

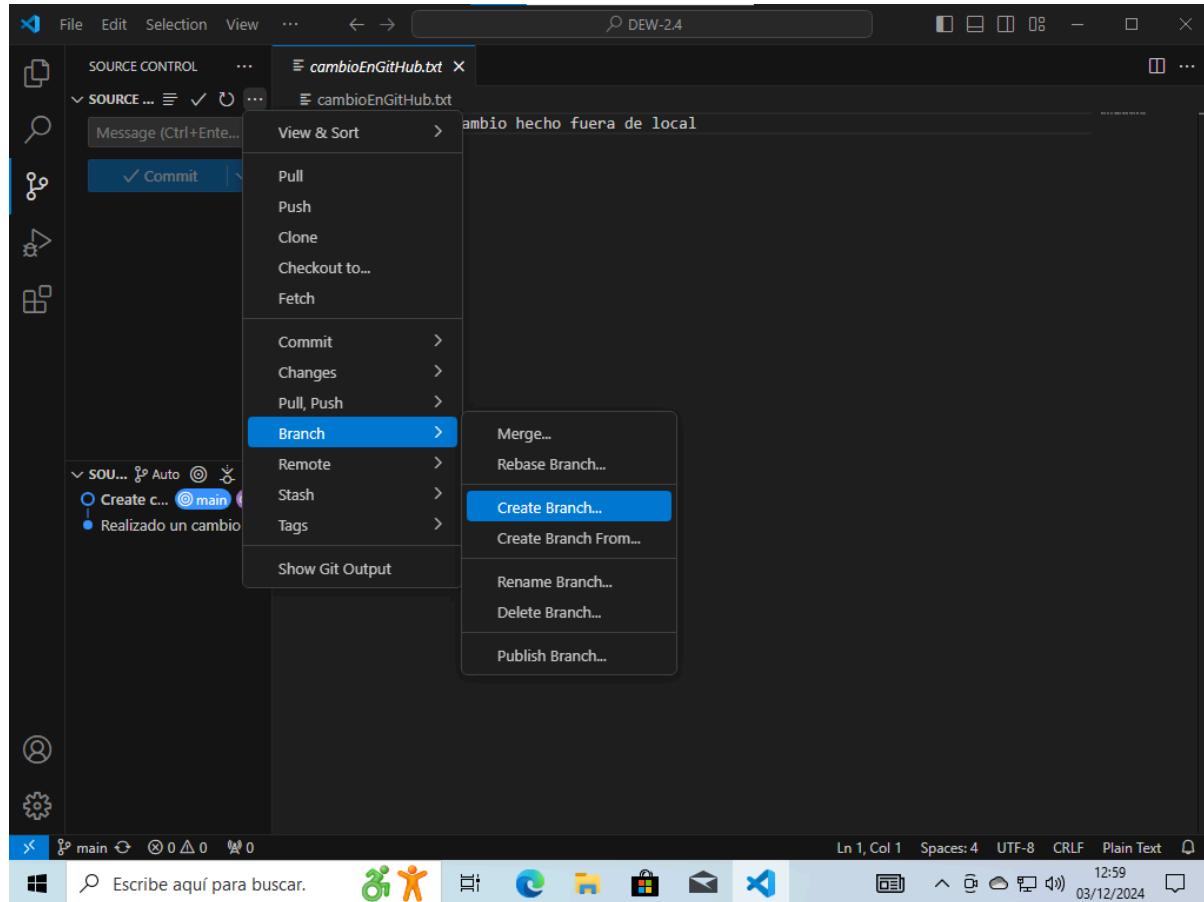


Tras haberlo hecho podemos comprobar el éxito del pull.

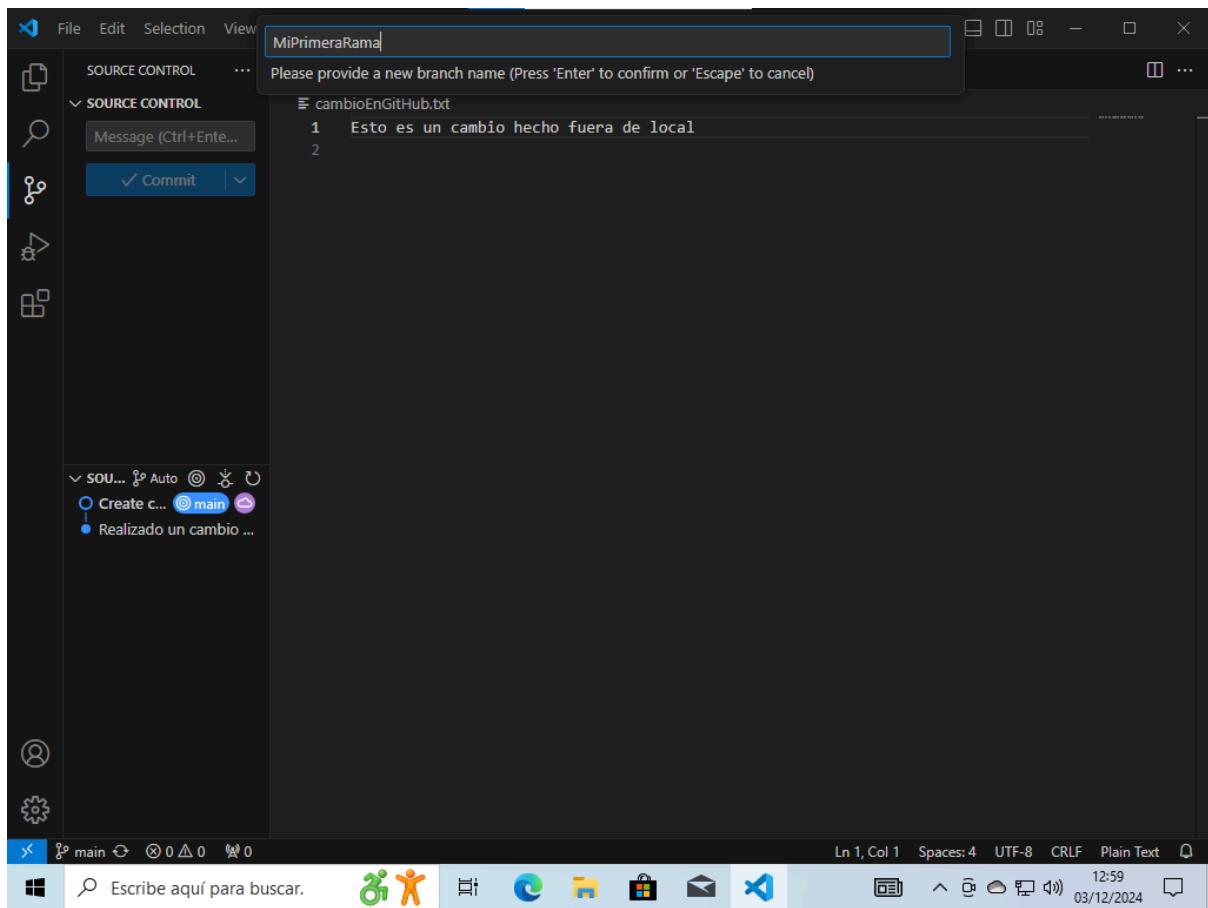


Crear una rama

Nos dirigimos al menú de control de versiones, en el menú de opciones, desplegamos el apartado de “Branch” y pulsamos “Create Branch”.



Nos pedirá un nombre para la rama y posteriormente tendremos creada nuestra rama en local.



Ahora debemos subirla a nuestro GitHub pulsando “Publish Branch”.

The screenshot shows a dark-themed interface of a code editor or terminal window. At the top, there's a navigation bar with icons for File, Edit, Selection, View, and a search bar labeled 'DEW-2.4'. Below the navigation bar is a toolbar with various icons. The main area contains a code editor window titled 'cambioEnGitHub.txt'. Inside the editor, the following text is displayed:

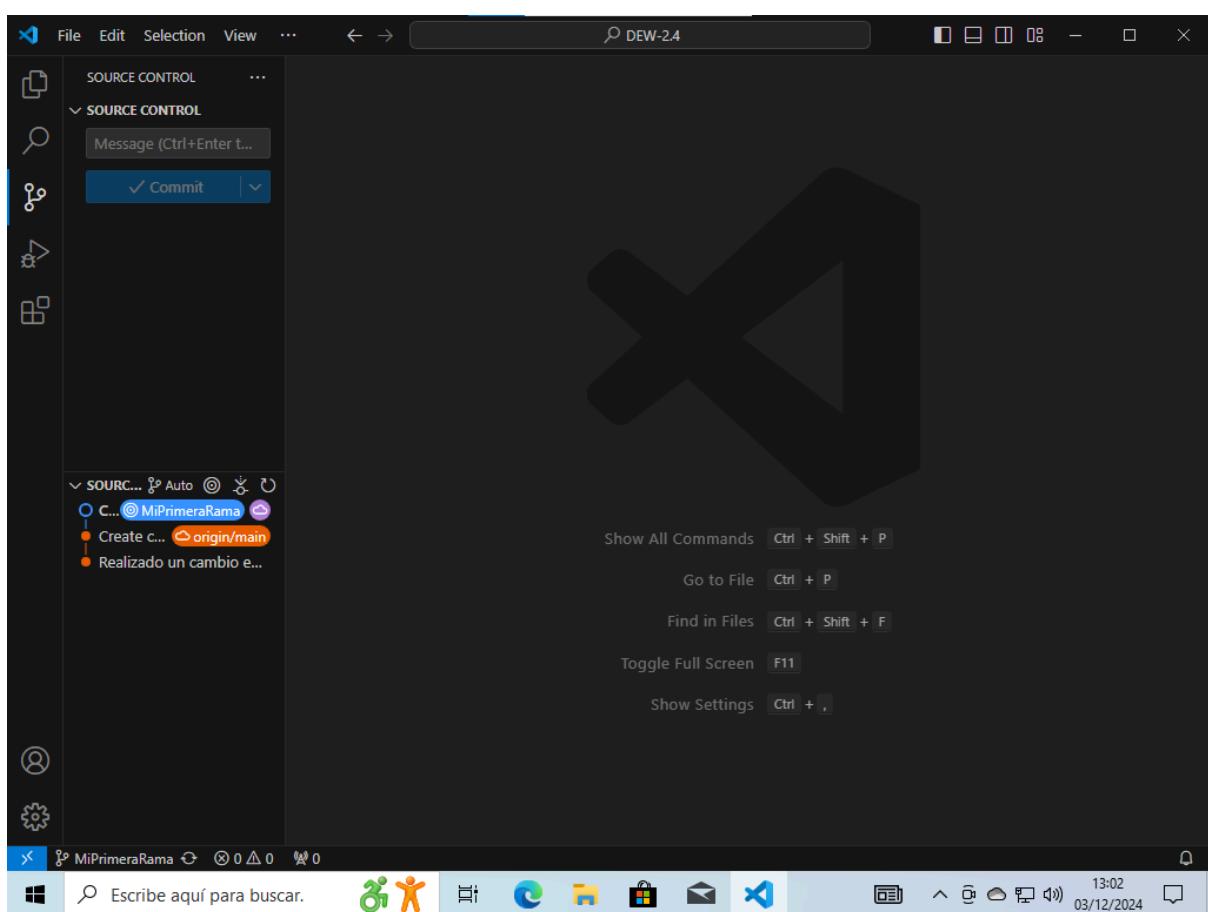
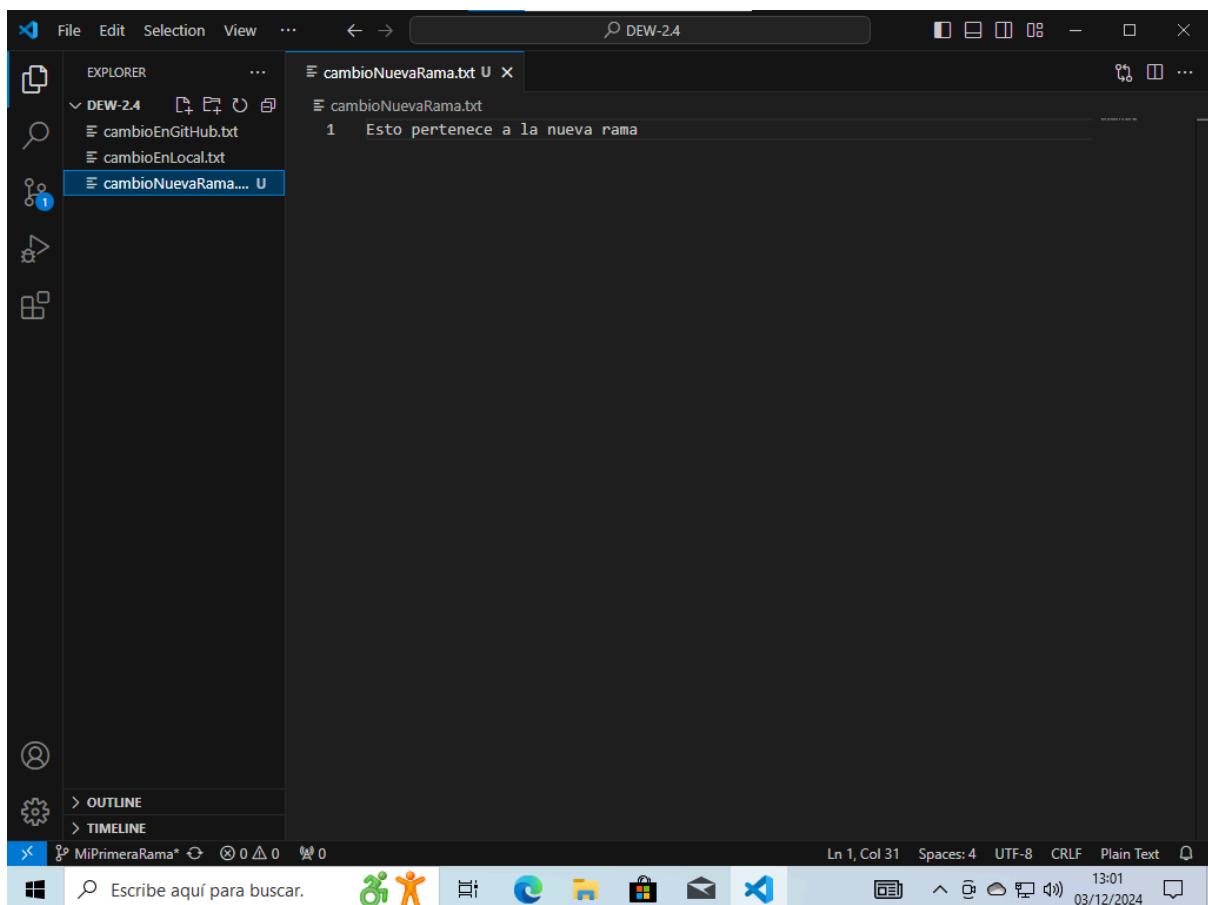
```
1 Esto es un cambio hecho fuera de local
2
```

Below the editor, a source control sidebar shows a single branch named 'MiPrimeraRama' with a status message: 'Realizado un cambio ...'. The bottom of the screen features a taskbar with several pinned application icons.

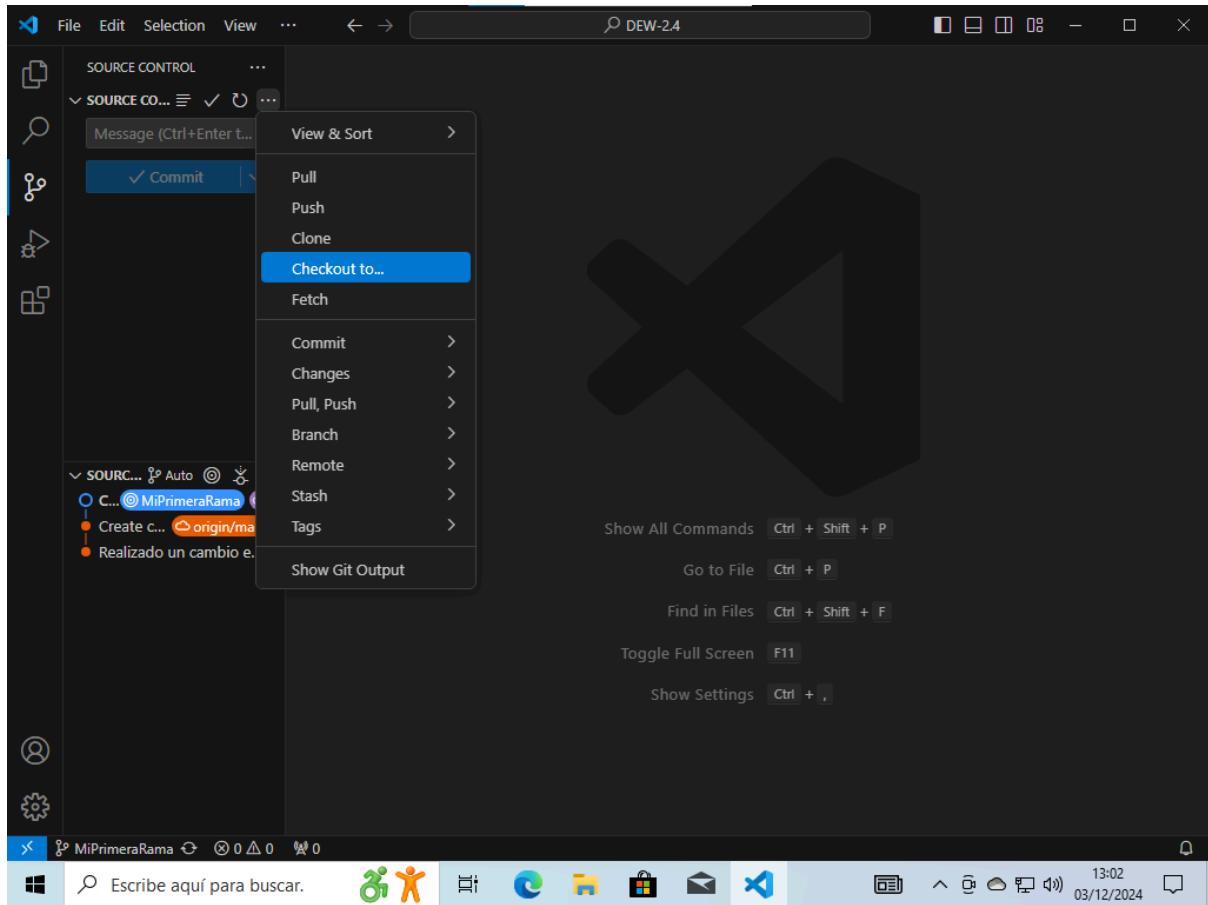
This screenshot is identical to the one above, except for the addition of a blue commit button at the bottom of the source control sidebar. The commit button has a checkmark icon and the text '✓ Commit'. The rest of the interface, including the code editor content and the source control status, remains the same.

Cambiar entre ramas

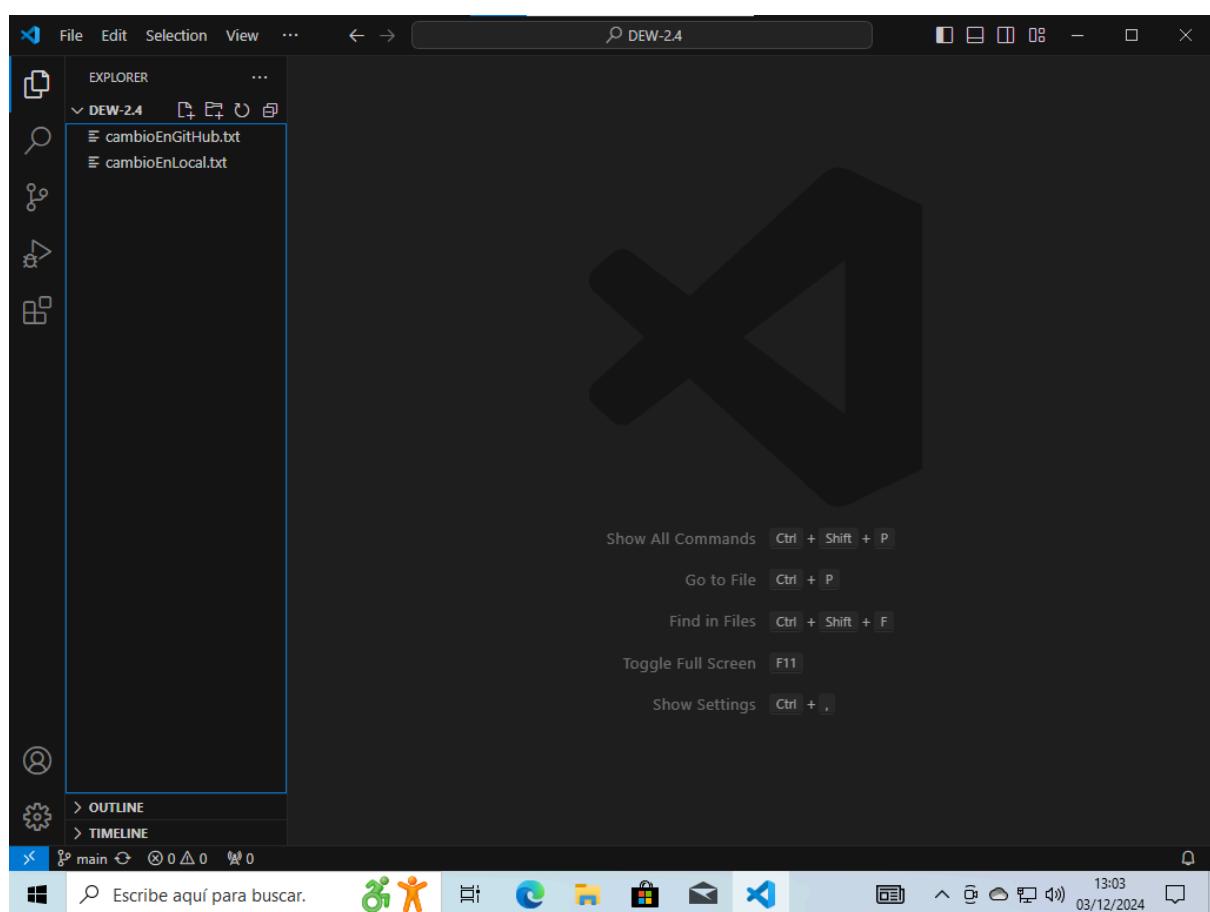
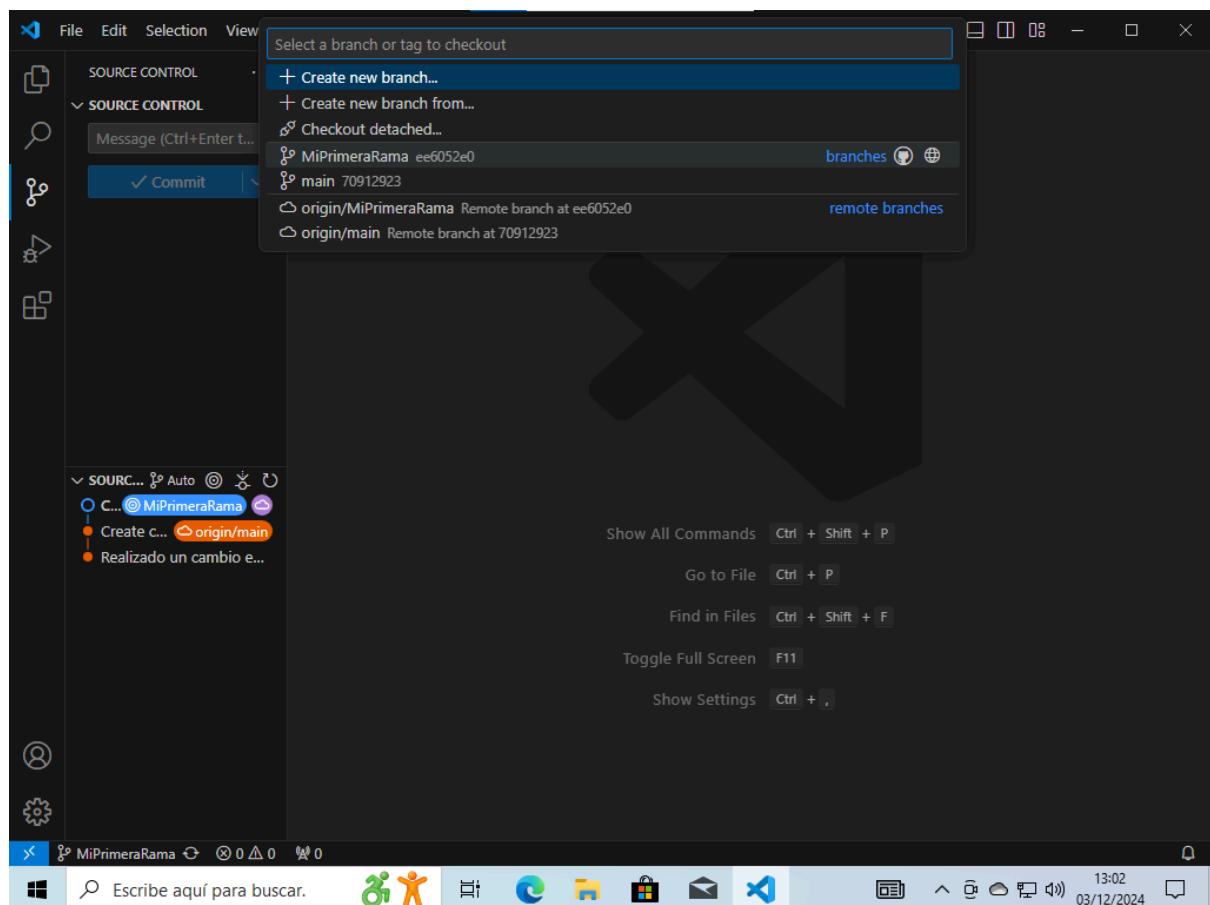
Primero generamos una diferencia entre las ramas para poder confirmar las ramas.



Ahora para cambiar a “Main” debemos dirigirnos al apartado de control de versiones, abrimos el menú de opciones y pulsamos “Checkout to”.

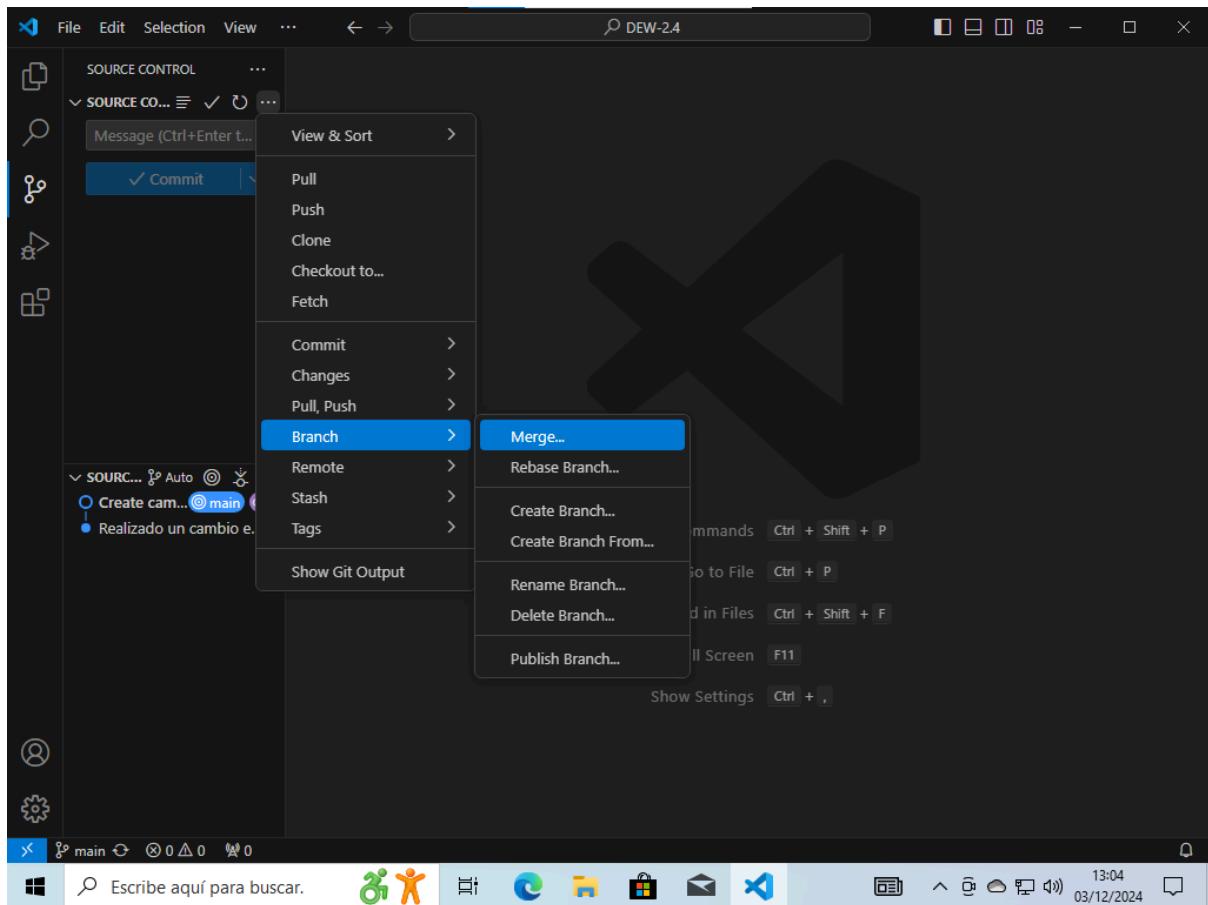


Aquí también podremos crear ramas, pero ahora lo que haremos es seleccionar la rama a la que queremos cambiar. Tras esto ya habremos cambiado de rama.

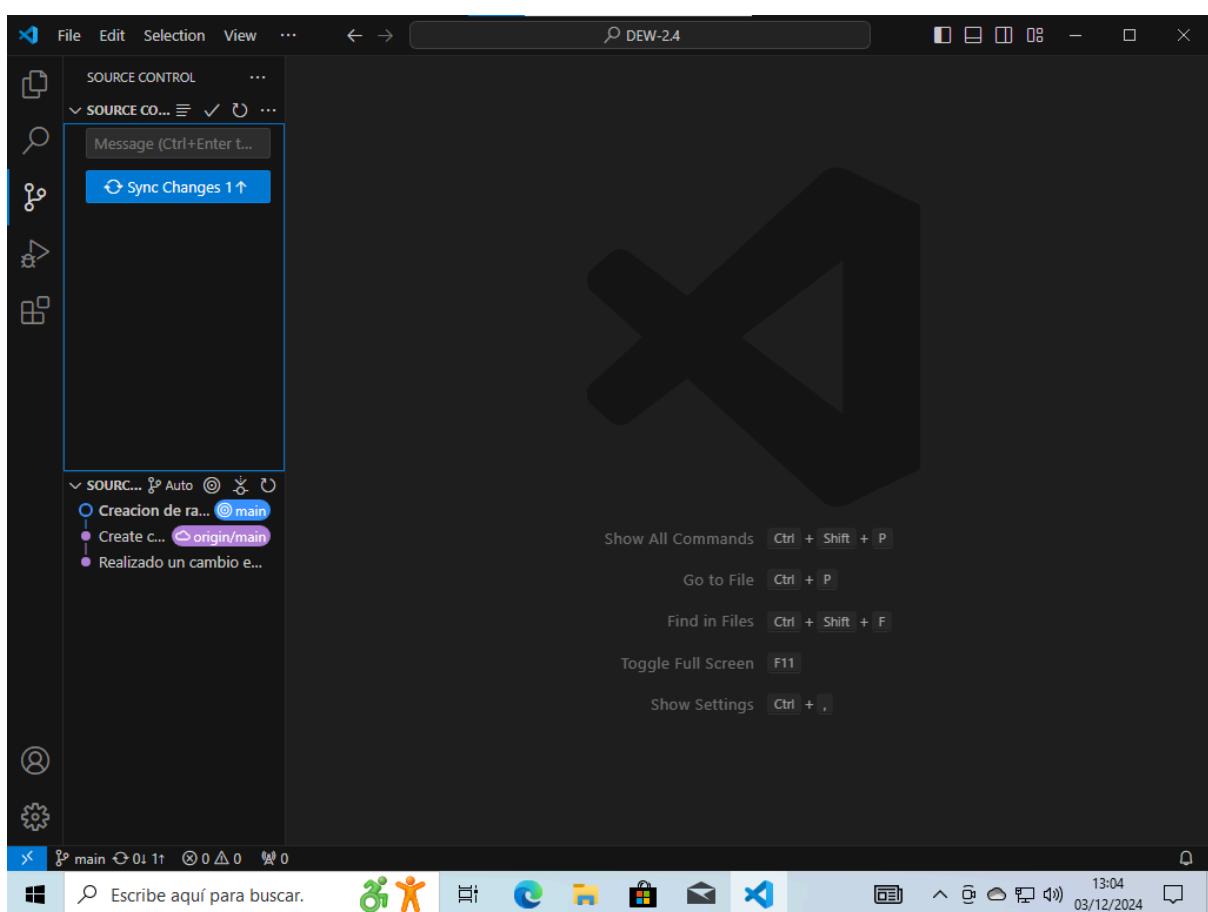
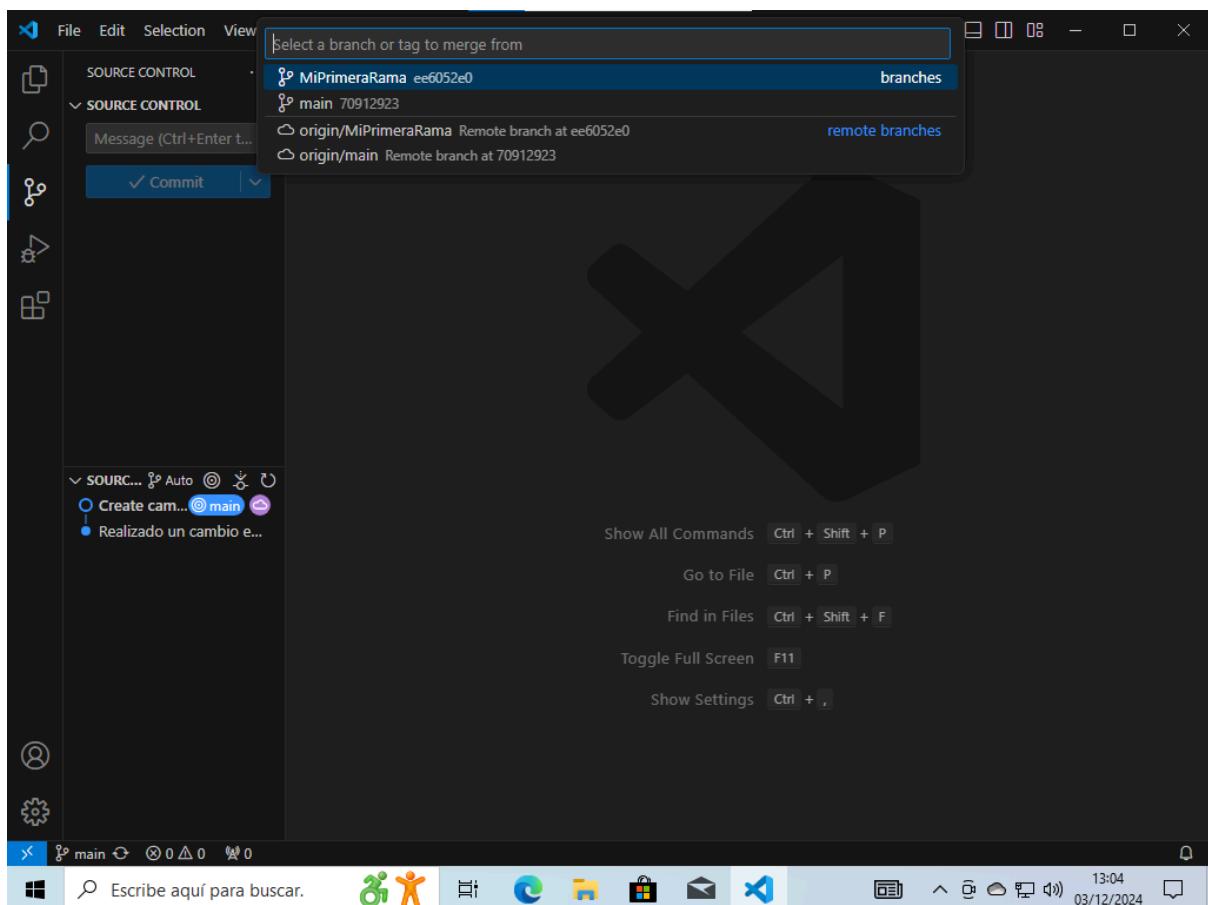


Merge de ramas

Nos dirigimos al apartado de control de versiones, desplegamos el menú de opciones, desplegamos el apartado de “Branch” y seleccionamos “Merge”.



Seleccionamos la rama con la que queremos hacer merge y posteriormente pulsamos “Sync Changes”.



Ahora podemos comprobar que el archivo que había en la rama secundaria también está en “Main”.

The screenshot shows a Microsoft Edge browser window with the following details:

- Title Bar:** Página principal - Google Drive | Instalación y uso de Git en Visual Studio | MJSDaw/DEW-2.4
- Address Bar:** https://github.com/MJSDaw/DEW-2.4/tree/main
- Repository Header:** MJSDaw / DEW-2.4 (Public)
- Code Tab:** Selected (highlighted in green).
- Commits List:**
 - MJSDaw Creacion de rama ee6052e · 4 minutes ago 3 Commits
 - cambioEnGitHub.txt Create cambioEnGitHub.txt 8 minutes ago
 - cambioEnLocal.txt Realizado un cambio en el local. 13 minutes ago
 - cambioNuevaRama.txt Creacion de rama 4 minutes ago
- About Section:** No description, website, or topics provided.
 - Activity
 - 0 stars
 - 1 watching
 - 0 forks
- Releases Section:** No releases published. [Create a new release](#)
- Packages Section:** No packages published. [Publish your first package](#)
- README Section:** README file is present but empty.
- Add a README:** Help people interested in this repository understand your project by adding a README.
- Search Bar:** Escribe aquí para buscar.
- Taskbar:** Icons for File Explorer, OneDrive, Microsoft Edge, Microsoft Store, Mail, and Task View.
- System Bar:** 13:06, 03/12/2024.

Resolver conflictos

Para resolver un conflicto generamos uno, en este caso cambiando código que estará diferente en el local.

The screenshot shows a GitHub code review interface. The repository is 'DEW-2.4/cambioNuevaRama.txt'. The commit message is 'Update cambioNuevaRama.txt' by 'fbee9ab · now'. The file content is:

```
1 Esto es un cambio para generar conflicto
```

The screenshot shows the 'DEW-2.4' project in Visual Studio Code. The file 'cambioNuevaRama.txt' is open, displaying the same content as the GitHub commit:

```
1 Esto es otro cambio en el mismo sitio para generar conflicto
```

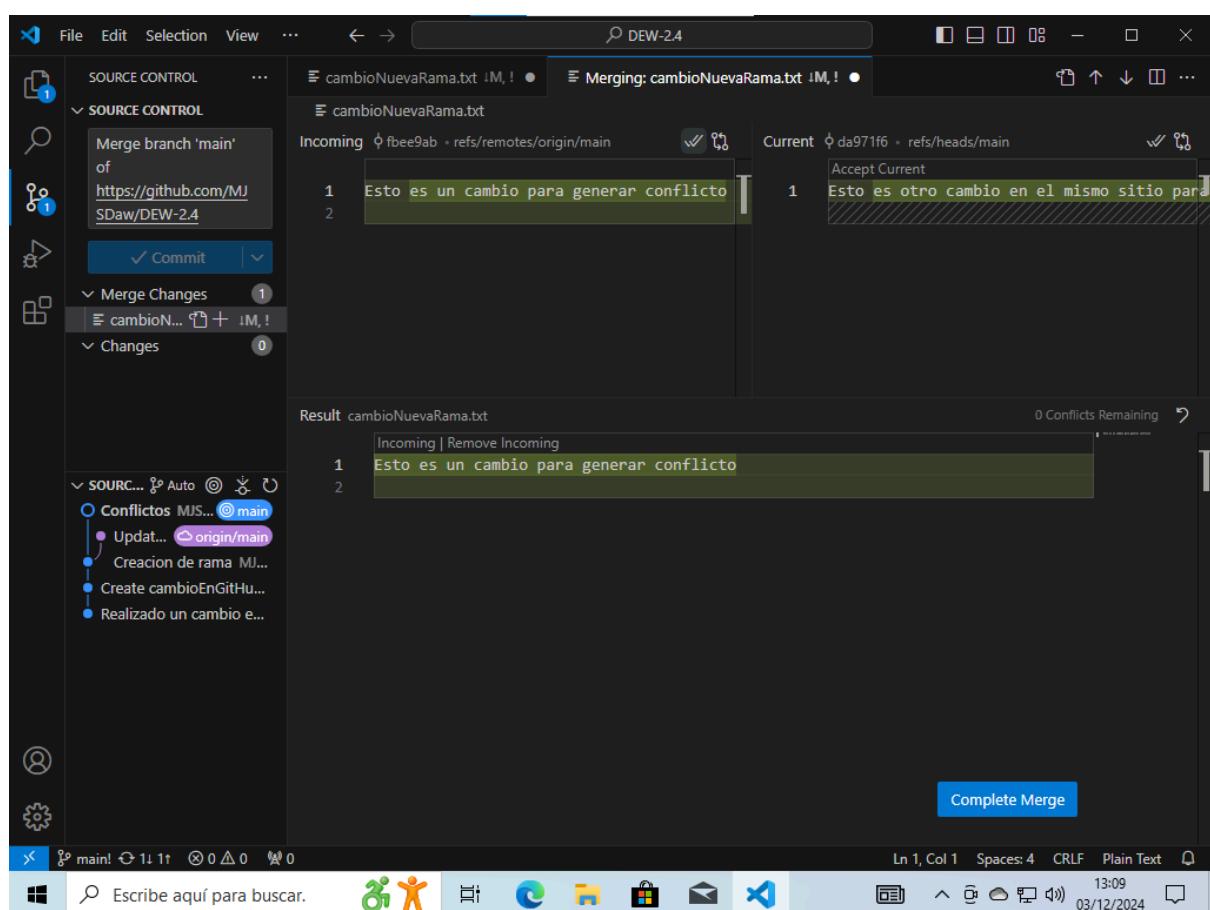
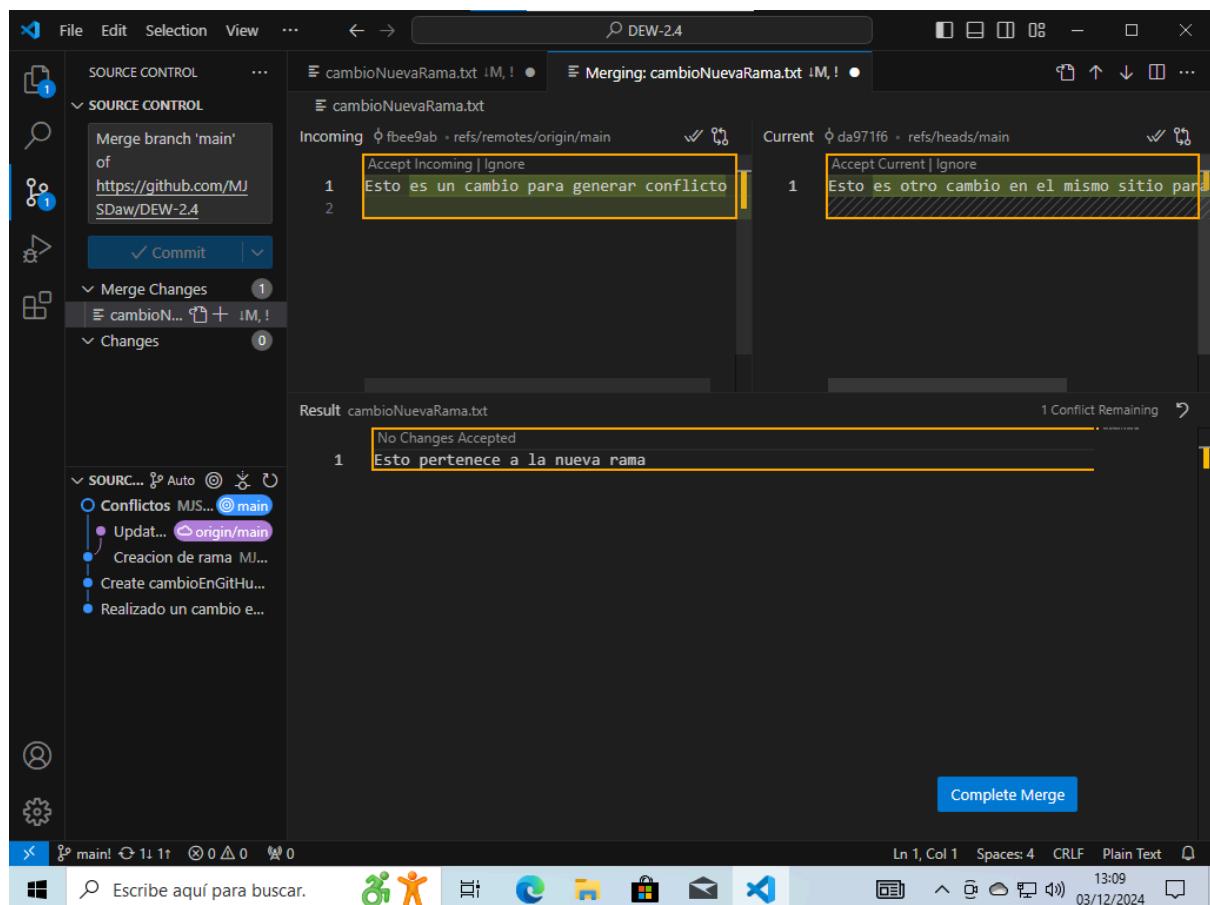
Ahora al hacer un pull podremos ver una ventana referente al conflicto. En la parte con <<<< estará el local y en la parte de >>>> estará el GitHub. Para proceder con la resolución pulsaremos el botón azul abajo a la derecha “Resolve in Merge Editor”.

The screenshot shows the GitHub Desktop application interface. On the left, there's a sidebar titled 'SOURCE CONTROL' with sections for 'Merge branch 'main'' and 'Changes'. The main area displays a file named 'cambioNuevaRama.txt' with the following content:

```
Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
<<<< HEAD (Current Change)
1 Esto es otro cambio en el mismo sitio para generar conflicto
2 =====
3 Esto es un cambio para generar conflicto
4 >>>> fbee9ab728e5891e0cbd05194c90f8c526498b58 (Incoming Change)
5
6
```

At the bottom right of the editor window, there is a blue button labeled 'Resolve in Merge Editor'.

En la derecha el código local, a la izquierda el código en GitHub, abajo lo que quieras que sea el resultado final. Una vez decidido pulsaremos “Complete Merge”.



Posteriormente haremos un commit para almacenar el resultado.

File Edit Selection View ... ← → 🔍 DEW-2.4

SOURCE CONTROL

Merge branch 'main' of https://github.com/MJSDaw/DEW-2.4

✓ Commit

Staged Changes 1

cambioNuevaRama.txt

Changes 0

SOURCE CONTROL

- Conflictos MJS... @ main
- Updat... origin/main
- Creacion de rama MJ...
- Create cambioEnGitHub...
- Realizado un cambio e...

main+ ↻ 1↑ 1↓ ⌂ 0 △ 0 ⌂ 0

Escribe aquí para buscar. 🔎

Ln 2, Col 1 Spaces: 4 UTF-8 CRLF Plain Text 13:09 03/12/2024

File Edit Selection View ... ← → 🔍 DEW-2.4

SOURCE CONTROL

Message (Ctrl+Enter t...)

✓ Commit

SOURCE CONTROL

- Merge br... @ main
- Update cambioNueva...
- Conflictos MJSDAW
- Creacion de rama MJ...
- Create cambioEnGitHub...
- Realizado un cambio e...

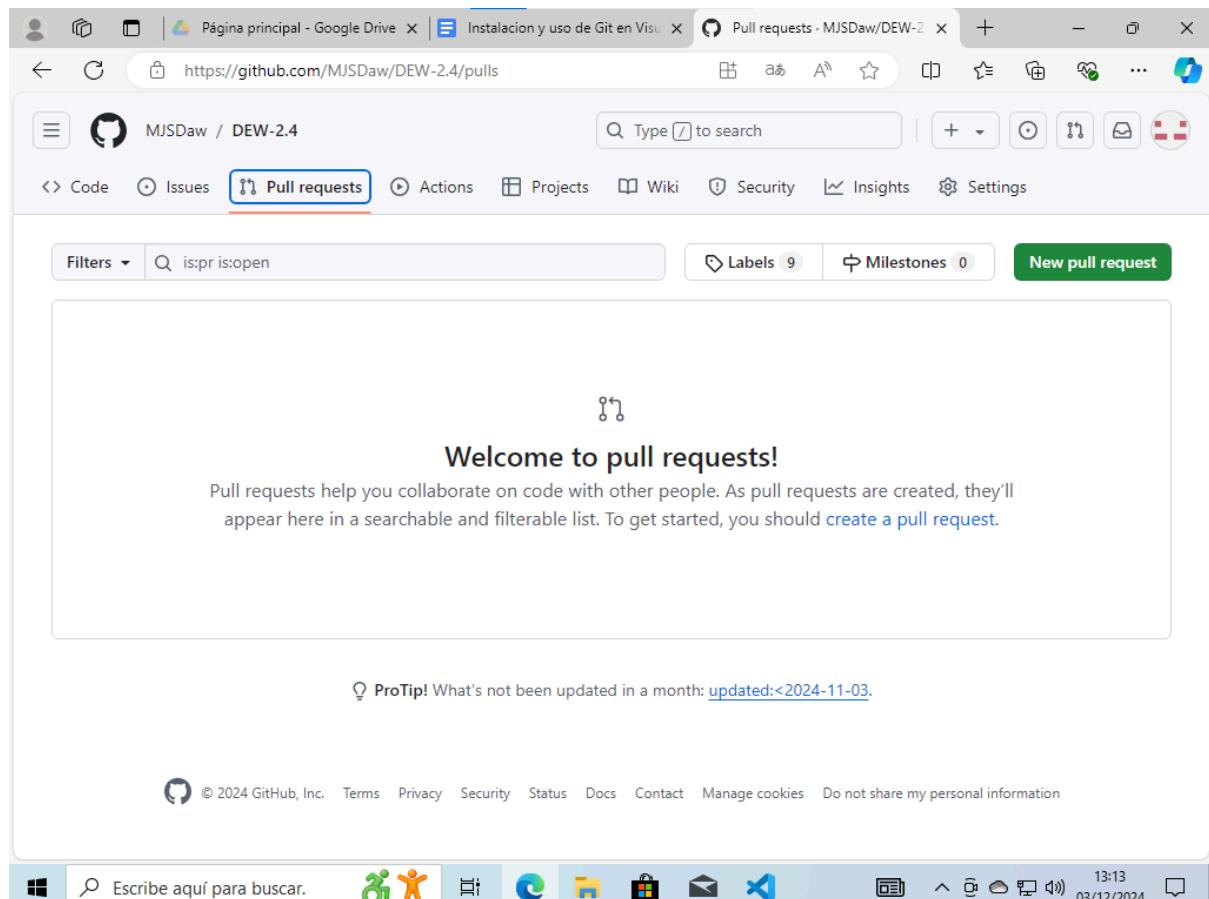
main ↻ 1↑ 1↓ ⌂ 0 △ 0 ⌂ 0

Escribe aquí para buscar. 🔎

Ln 2, Col 1 Spaces: 4 UTF-8 CRLF Plain Text 13:10 03/12/2024

Crear Pull Request

Para crear un “Pull Request” nos iremos a la página oficial de GitHub, al repositorio donde queremos crearlo y al apartado de “Pull Request”. Ahí veremos los diferentes requests y pulsando “New Pull Request” crearemos uno.



Seleccionaremos la rama “base” donde se aplicará el merge (el resultado de la rama unificada) y en “compare” seleccionamos la rama que queremos unir. Posteriormente pulsamos “Create pull request”.

Compare changes

Compare changes across branches, commits, tags, and more below. If you need to, you can also [compare across forks](#).

base: main ⌂ compare: main ⌂ ...

Choose different branches or forks above to discuss and review changes. [Learn about pull requests](#)

Create pull request

Example comparisons

MiPrimeraRama	12 minutes ago
main@{1day}...main	24 hours ago

© 2024 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact](#) [Manage cookies](#) [Do not share my personal information](#)

Escribe aquí para buscar.

Añadiremos un título, descripción y, en la parte de la derecha, podemos seleccionar un “Reviewer” para que revise el cambio y que aceptara el merge o no.

The image displays two side-by-side screenshots of a GitHub pull request creation interface, illustrating the process of merging one branch into another.

Screenshot 1 (Top):

- Title:** MiPrimeraRama...main
- Description:** Add your description here... (with a note: "Markdown is supported")
- Reviewers:** No reviews
- Assignees:** No one (with a link to "assign yourself")
- Labels:** None yet
- Projects:** None yet
- Milestone:** No milestone
- Development:** Use [Closing keywords](#) in the description to automatically close issues
- Helpful resources:** [GitHub Community Guidelines](#)

Screenshot 2 (Bottom):

- Title:** Comparing MiPrimeraRama...main
- Description:** Merge de una rama a otra (with a note: "asdasdasdasdasd")
- Reviewers:** Request up to 15 reviewers (with a "Type or choose a user" input field)
- Assignees:** Nothing to show
- Labels:** None yet
- Projects:** None yet
- Milestone:** No milestone
- Development:** Use [Closing keywords](#) in the description to automatically close issues
- Helpful resources:** [GitHub Community Guidelines](#)

Common UI Elements:

- Header:** Página principal - Google Drive | Instalacion y uso de Git en Visu | Comparing MiPrimeraRama...main
- Toolbar:** Back, Forward, Home, Refresh, etc.
- Search Bar:** Escribe aquí para buscar.
- System Icons:** Taskbar icons for File Explorer, Task View, Edge, File History, Mail, and others.
- Date/Time:** 13:15 03/12/2024

Ya podremos ver en el apartado de “Pull requests” el que acabamos de crear.

The screenshot shows a Microsoft Edge browser window with the following details:

- Title Bar:** Página principal - Google Drive | Instalación y uso de Git en Visual Studio | Pull requests · MJSDaw/DEW-2
- Address Bar:** https://github.com/MJSDaw/DEW-2.4/pulls
- Page Content:**
 - Header:** MJSDaw / DEW-2.4, Issues, Pull requests 1 (highlighted), Actions, Projects, Wiki, Security, Insights, Settings
 - Search Bar:** Type to search
 - Filters:** Filters dropdown, is:pr is:open, Labels 9, Milestones 0, New pull request button
 - List:** 1 Open, 0 Closed. One item is listed:
 - Merge de una rama a otra** (checkbox)
 - #1 opened now by MJSDaw
 - Tip:** ProTip! Follow long discussions with comments:>50.
 - Footer:** GitHub logo, © 2024 GitHub, Inc., Terms, Privacy, Security, Status, Docs, Contact, Manage cookies, Do not share my personal information
- Taskbar:** Shows the Windows Start button, a search bar with placeholder "Escribe aquí para buscar.", and pinned icons for File Explorer, Edge, Task View, File History, Mail, and Microsoft Store.