

# **Instalación y Configuración de Herramienta de Versionamiento (Local / Web)**

ESTEBAN VALBUENA

SENA

ANALISIS DE DESARROLLO Y SOFTWARE

ACTIVIDAD: **GA7-220501096-AA1-EV04**

TUTOR: MAURICIO FERNANDEZ

**24/04/2024**

## **Introducción**

El control de versionamiento es una práctica fundamental en el desarrollo de software, ya que permite gestionar los cambios realizados en el código fuente de manera eficiente. Git es una de las herramientas más populares para este propósito, ofreciendo capacidades tanto para el control de versiones locales como remotas. Este documento detalla el proceso de instalación y configuración de Git en ambos contextos, proporcionando una guía paso a paso acompañada de pantallazos para facilitar la comprensión y replicación del proceso.

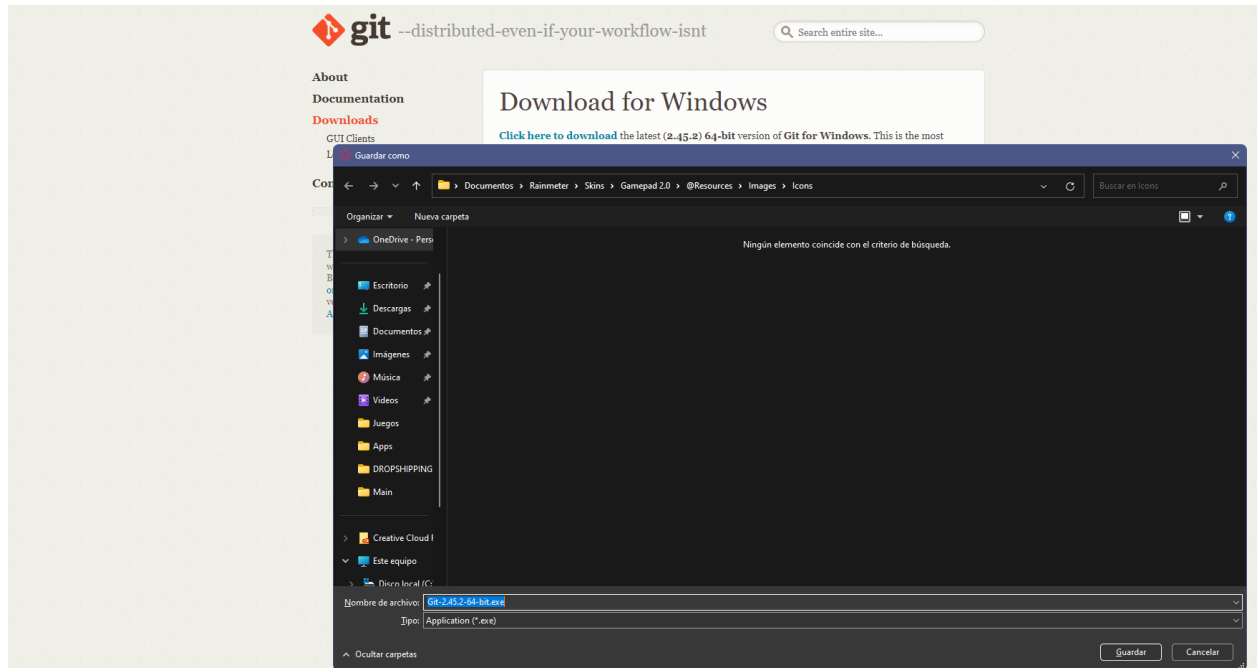
## **Objetivo**

El objetivo de este documento es guiar a los desarrolladores a través del proceso de instalación y configuración de Git, tanto en su entorno local como remoto. Al finalizar este documento, el lector debería ser capaz de instalar Git en su máquina local, configurar un repositorio local y conectar y configurar un repositorio remoto en una plataforma como GitHub.

## Paso a Paso

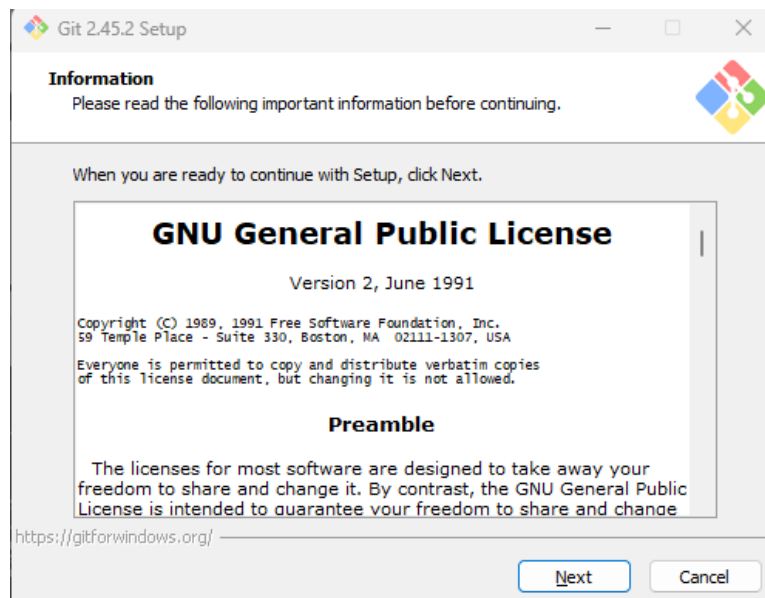
## Instalación y Configuración de Git Local

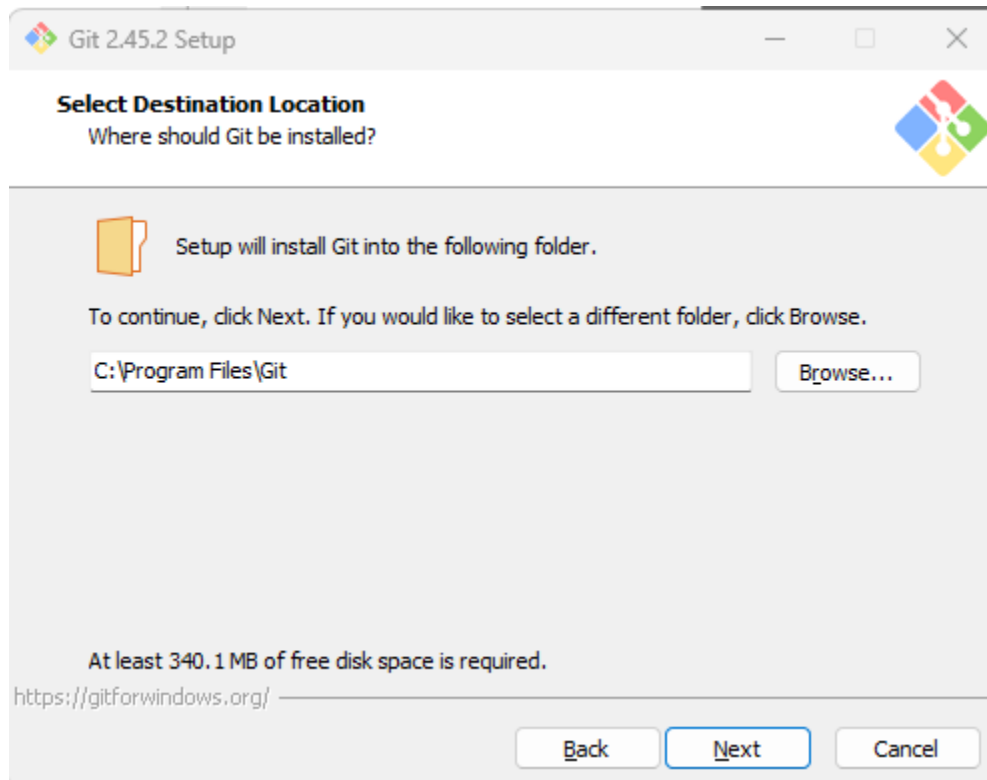
### Descargar git



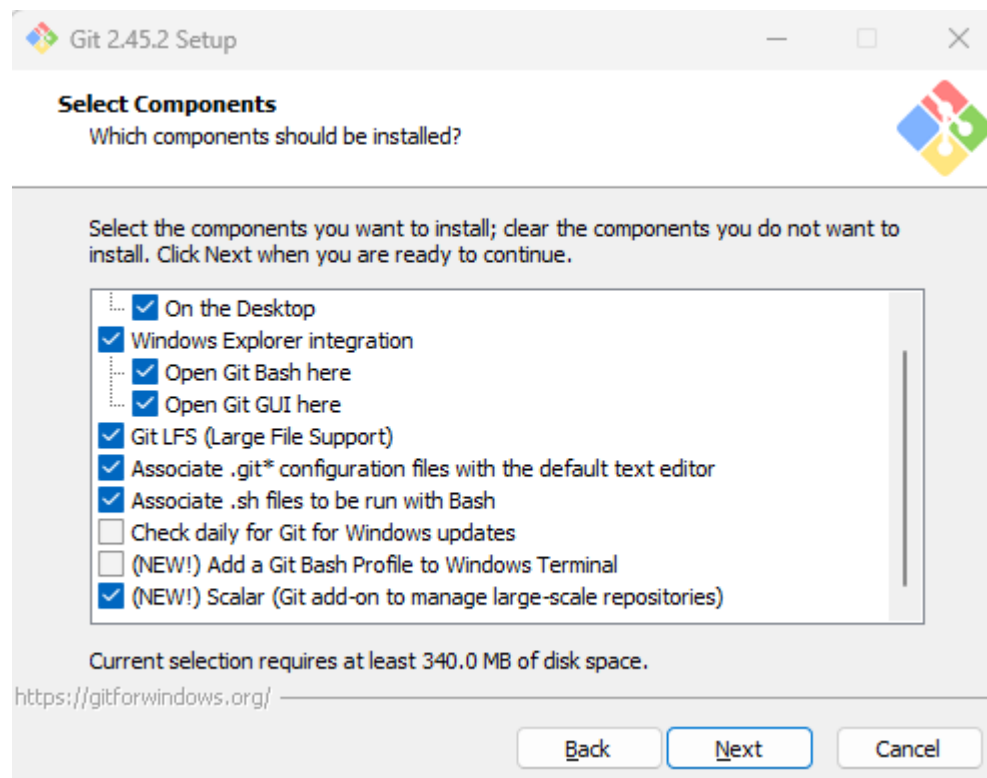
### Instalar Git

- Ejecutamos el archivo descargado para iniciar el instalador.
- Continuamos siguiendo las instrucciones del asistente de instalación

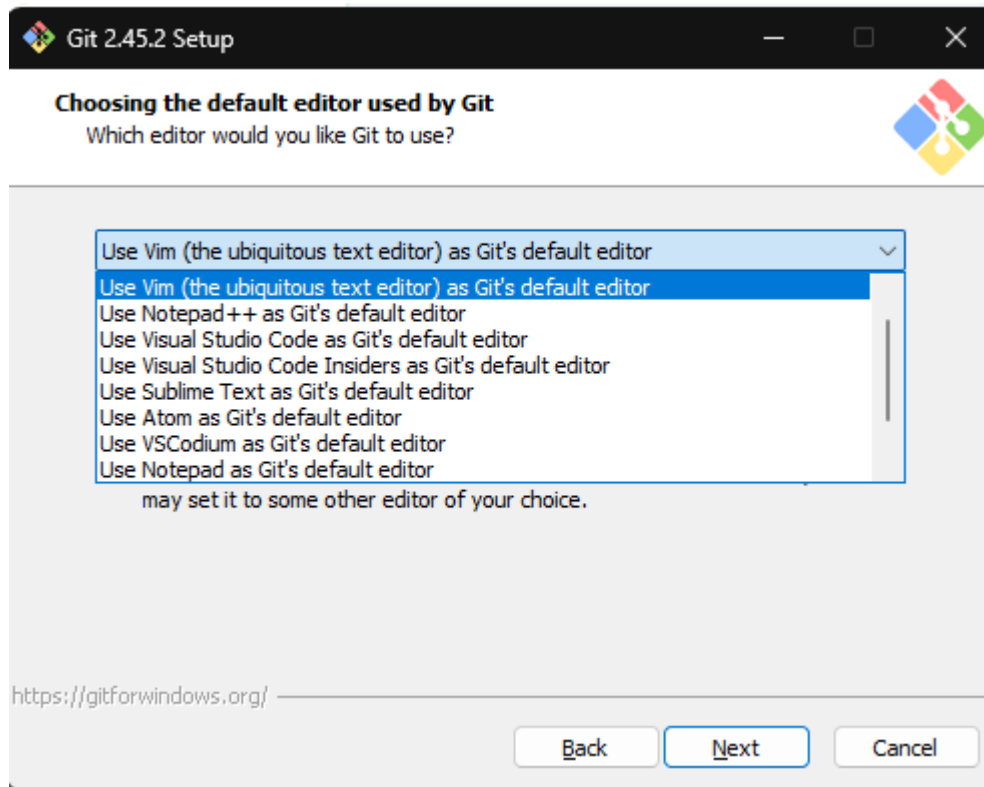




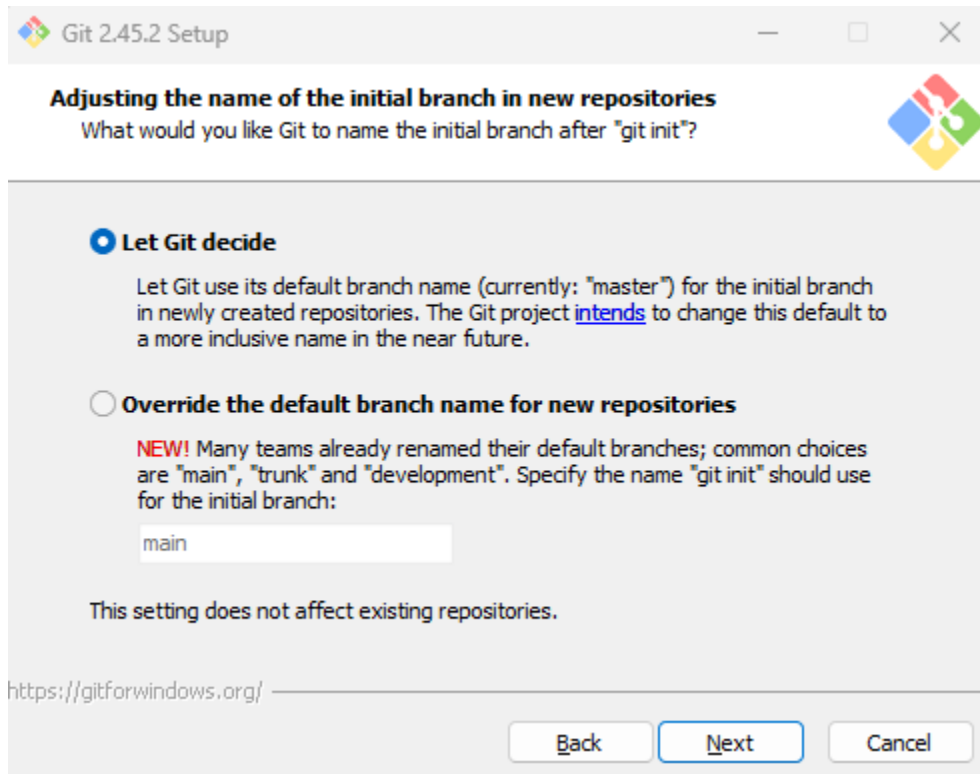
Hacemos selección de las preferencias que optemos por dejar o colocar



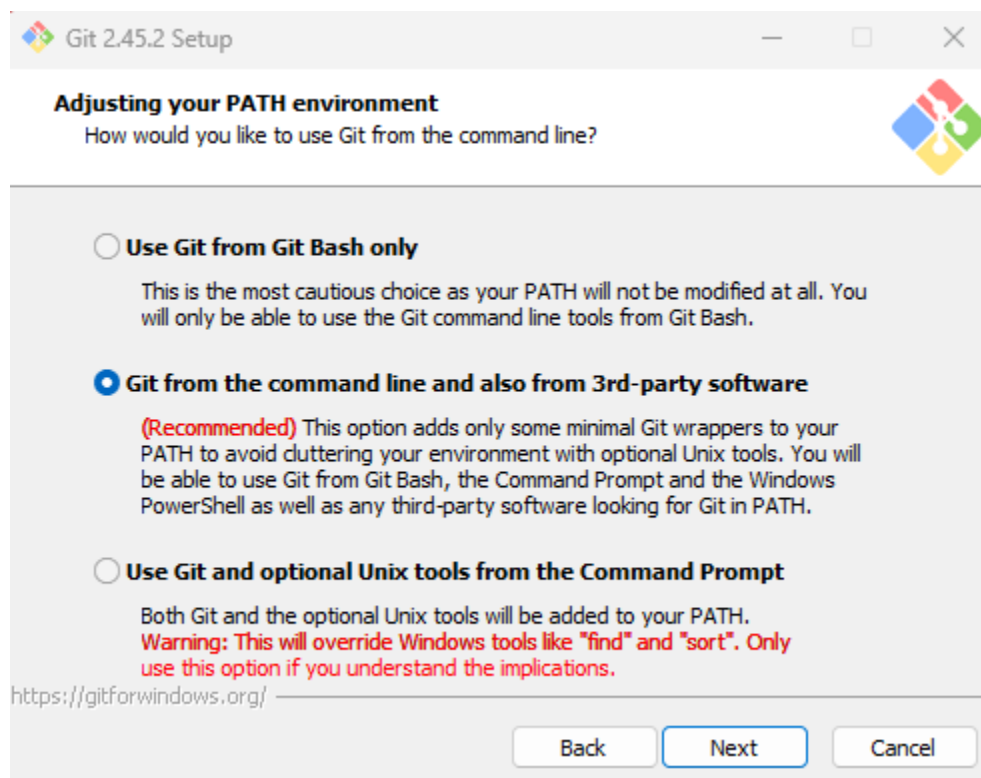
Seleccionamos el editor que mejor nos parezca, en mi caso dejar el predeterminado ya que es la primera vez que hacemos uso de **GIT**



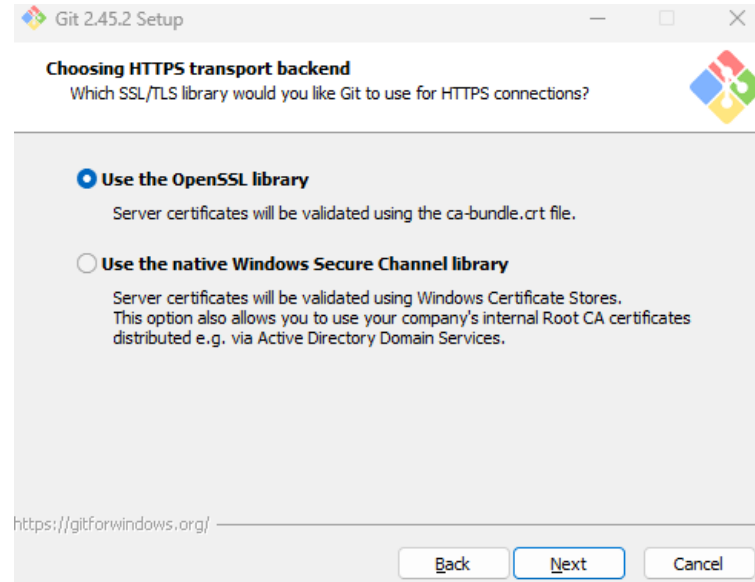
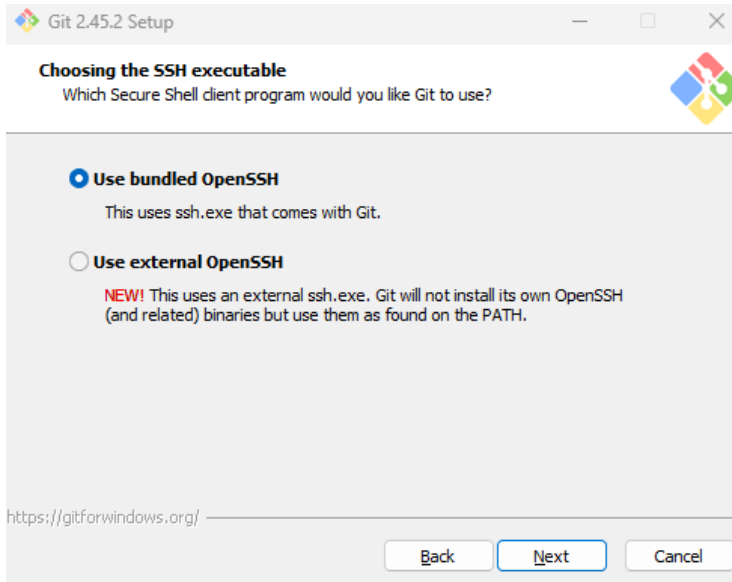
Dejemos aquí que **GIT** decida, ya que pregunta por el nombre que nos gustaría que se le asigne al demo



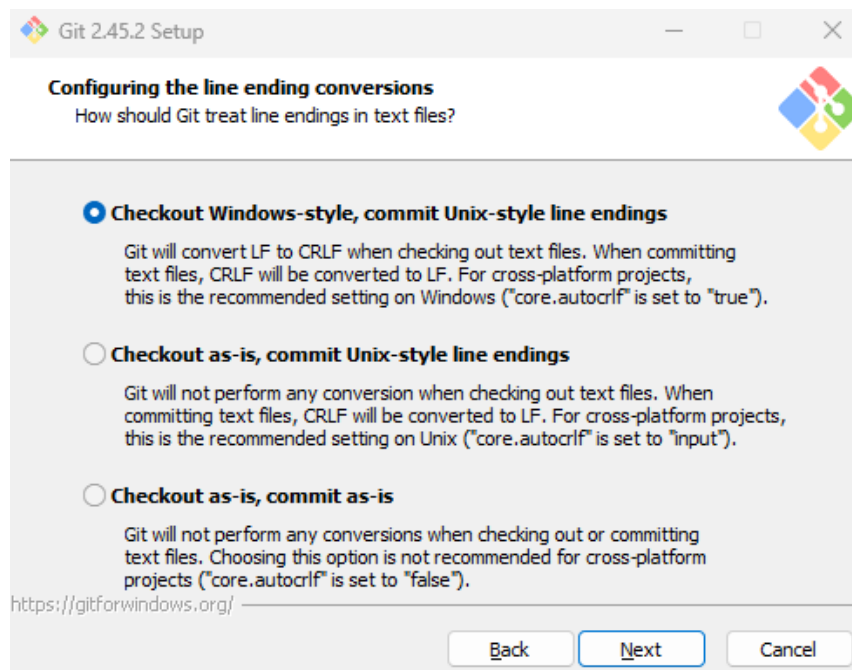
En este caso, se nos pregunta qué programas pueden usar GIT; la primera opción indica que solo GIT BASH, mientras que la tercera opción indica que otros programas también pueden usarlo; por lo tanto, elegiremos la segunda opción, que es la línea de comando y de terceros.



Aquí nos pregunta de qué librería queremos hacer uso para realizar comunicaciones remotas con repositorios externos, escogeré **OPENSSL** para iniciar de una forma eficiente

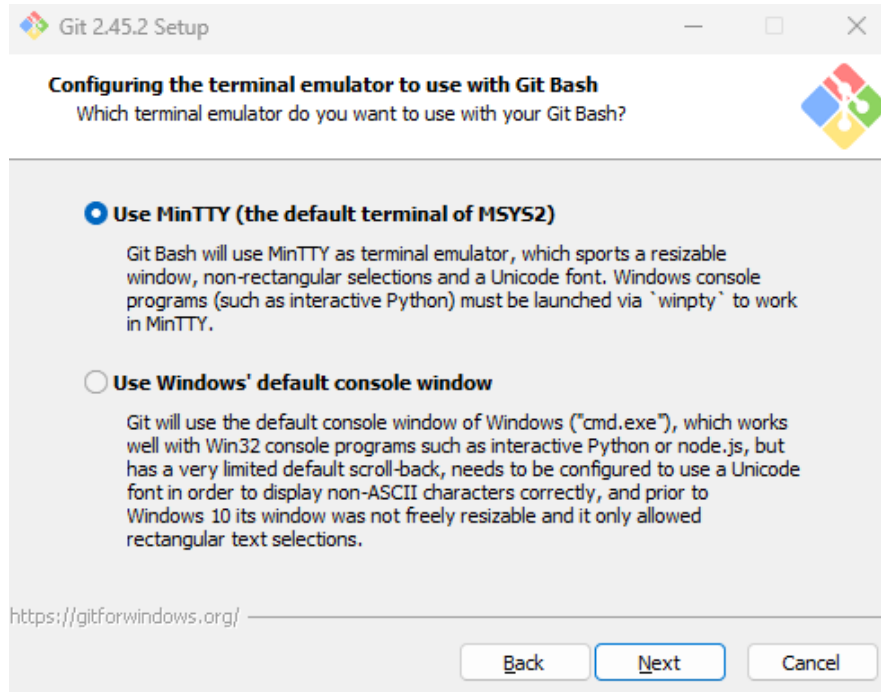


Aquí nos pregunta cómo queremos hacer frente a los saltos en línea, dejare por predeterminada la primera opción

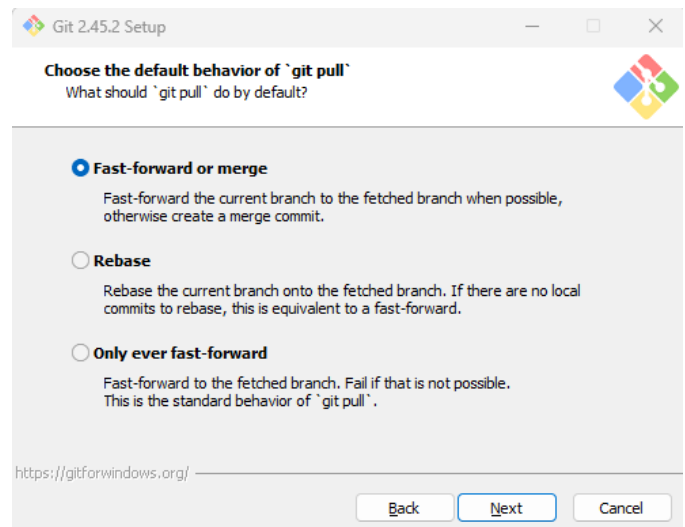
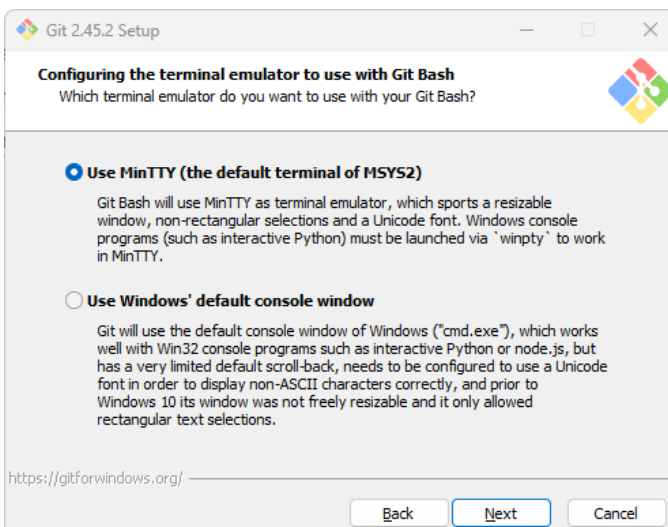




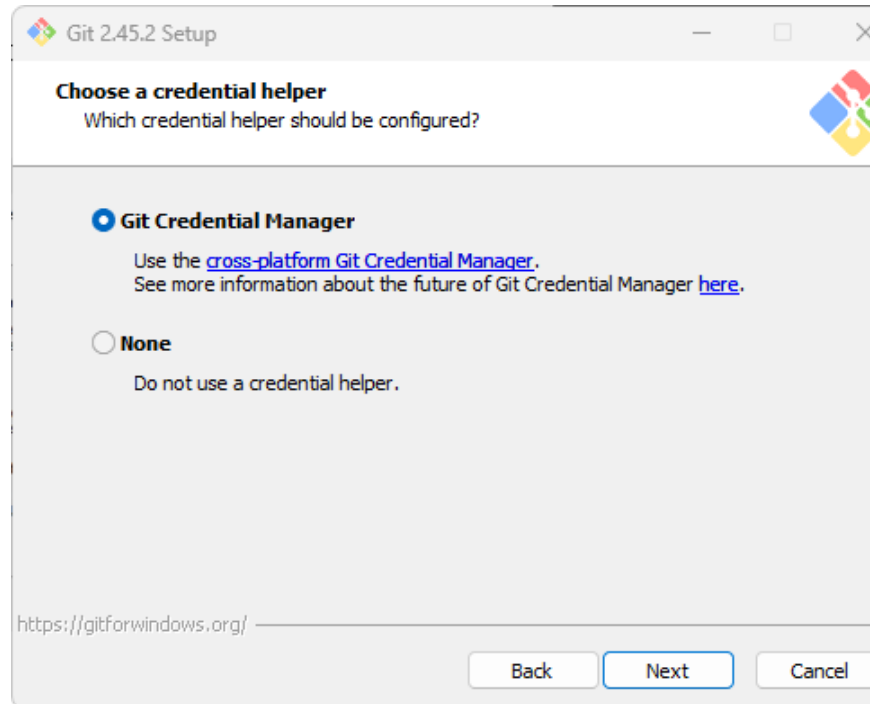
Aquí determinamos que terminal usaremos para poder ejecutar GIT, dejare la que está por predeterminado



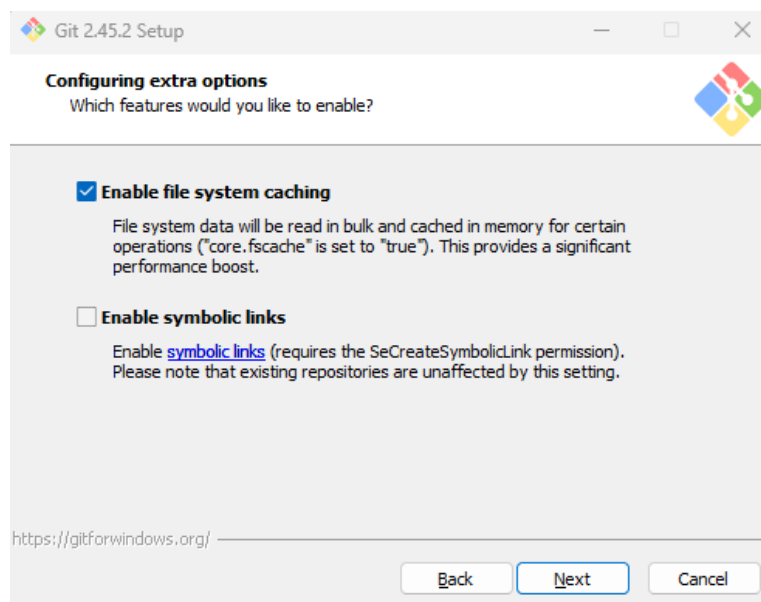
En esta sección pregunta sobre el comportamiento por defecto cuando hagamos git pull. El comando git pull nos permite extraer archivos o cambios en un repositorio remoto, pero como todavía no manejamos bien **GIT**, lo dejaremos por defecto.



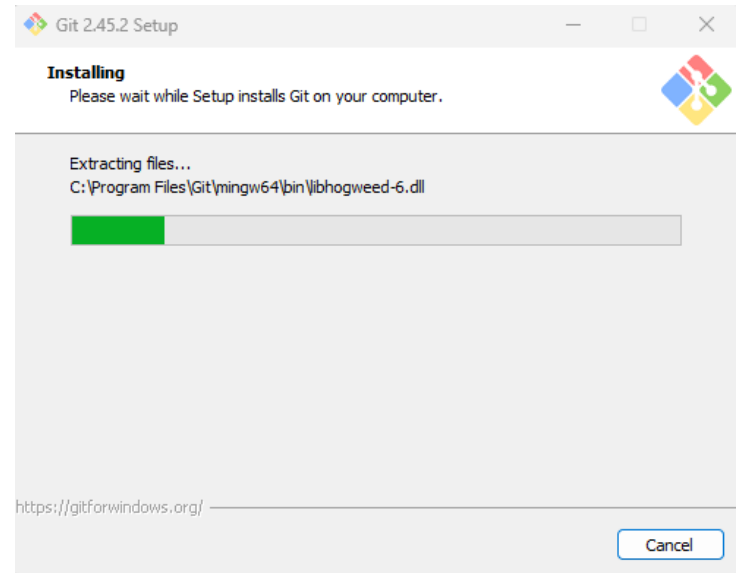
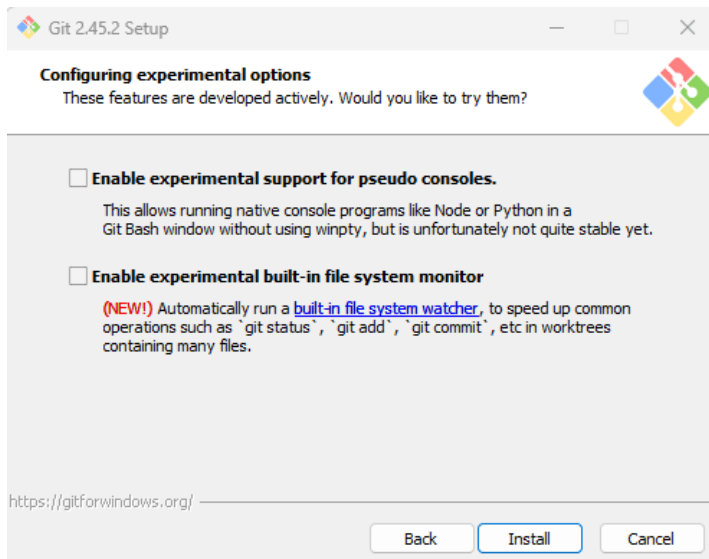
Aquí se nos pregunta para las credenciales del software, esto es para manejar nuestra cuenta de **GIT**, escogeremos la primera opción



Finalizando, indica si queremos habilitar solo los caches o links simbolicos, yo dejare la primera opción



Y por ultimo nos pregunta si deseamos hacer uso de configuraciones extras que están en fase de beta, como aun no nos relacionamos del todo con el software, simplemente le damos en instalar



## **Conclusión**

Este documento ha proporcionado una guía detallada para la instalación de Git en entornos locales y remotos. La capacidad de gestionar versiones de código de manera eficiente es crucial para cualquier proyecto de desarrollo de software, y el uso de herramientas como Git y plataformas como GitHub facilita enormemente este proceso. Siguiendo los pasos descritos, los desarrolladores pueden establecer un flujo de trabajo de control de versiones sólido y colaborativo.