



INSTALACIÓN Y CONFIGURACIÓN DE HERRAMIENTA DE VERSIONAMIENTO (LOCAL / WEB) GA7-220501096-AA1- EV04

Taller Práctico

MAYO DE 2024

SENA

Centro Ambiental y Ecoturístico del Nororiente Amazónico

Nombre LUIS FERNANDO PEÑALOZA ORTIZ (C.C 1216971980)
Análisis y desarrollo de software
(Ficha: 2547418)



INTRODUCCIÓN

La instalación y configuración de herramientas de versionamiento, ya sea en entornos locales o en la web, es una parte fundamental en el flujo de trabajo de cualquier equipo de desarrollo de software. Estas herramientas ofrecen un método estructurado y eficiente para gestionar el control de versiones del código fuente, facilitando la colaboración, el seguimiento de cambios y la gestión de proyectos.

En este contexto, el tema de la instalación y configuración de herramientas de versionamiento aborda la preparación inicial necesaria para adoptar un sistema de control de versiones en un proyecto de desarrollo de software. Esto implica la selección de la herramienta adecuada para las necesidades del equipo, su instalación en el entorno de desarrollo local y/o en plataformas en la nube, así como su configuración inicial para optimizar el flujo de trabajo del equipo.



Instalación y configuración de herramienta de versionamiento (Local / Web)

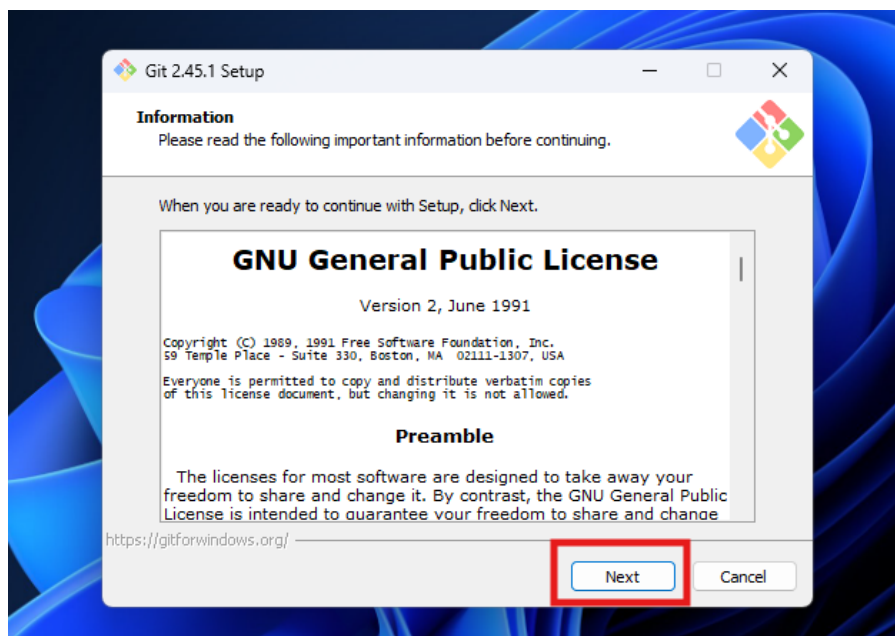
Descargar Git

Se accede al sitio oficial de Git (<https://git-scm.com/>) y se descarga el instalador adecuado para Windows.



Ejecutar el Instalador

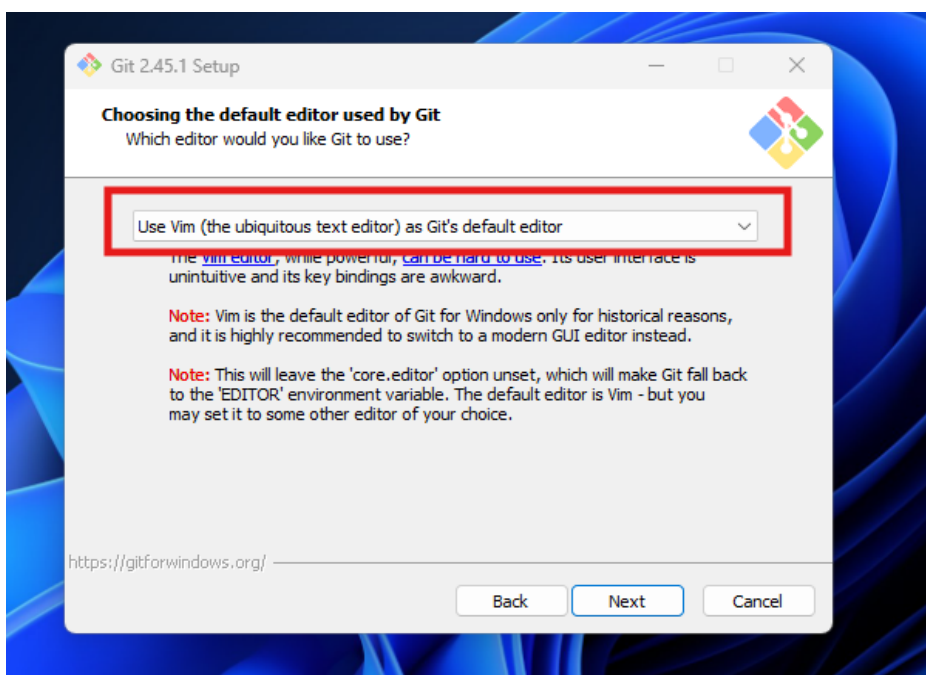
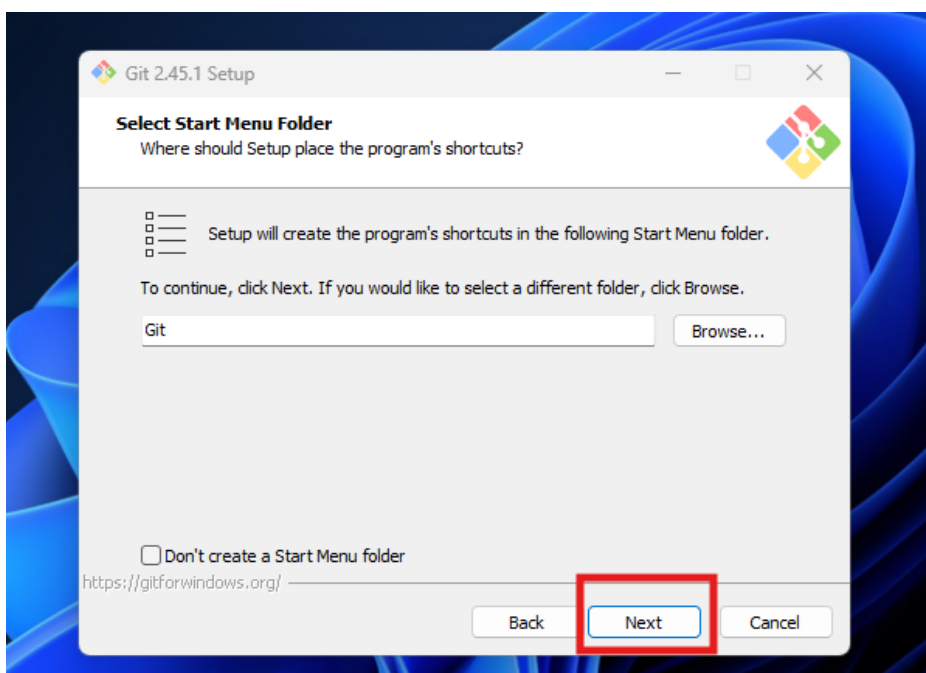
Se abre el archivo descargado y se inicia el asistente de instalación.





Configurar la Instalación

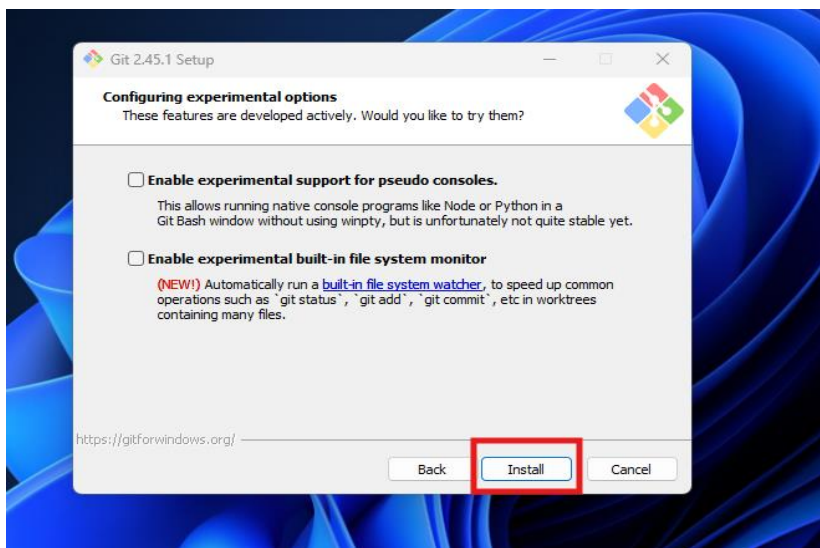
Se seleccionan las opciones predeterminadas sugeridas por el asistente, como la ubicación de instalación, componentes adicionales, editor predeterminado, ajustes del PATH, configuración de transporte HTTPS, estilo de final de línea y terminal de Git.





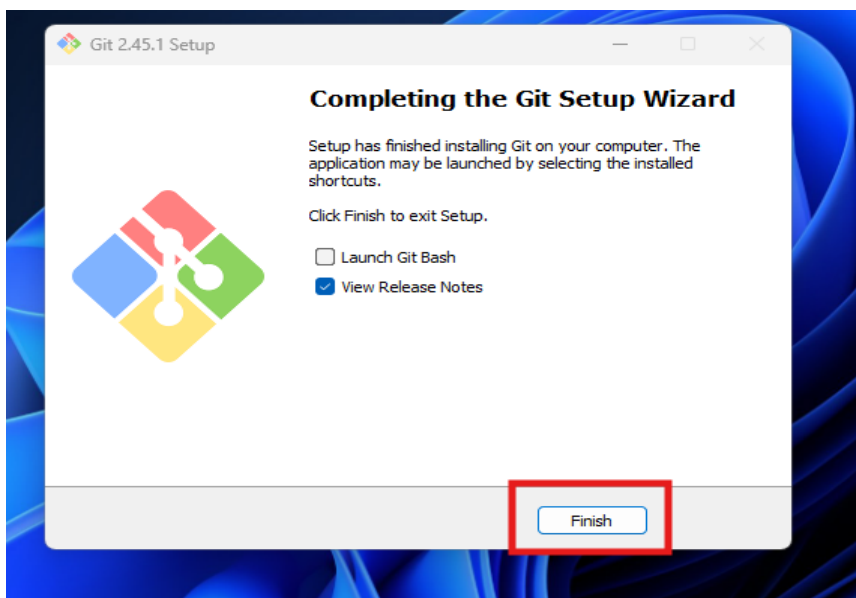
Completar la Instalación

Una vez finalizada la instalación, se verifica abriendo Git Bash y comprobando la versión instalada con el comando ``git --version``



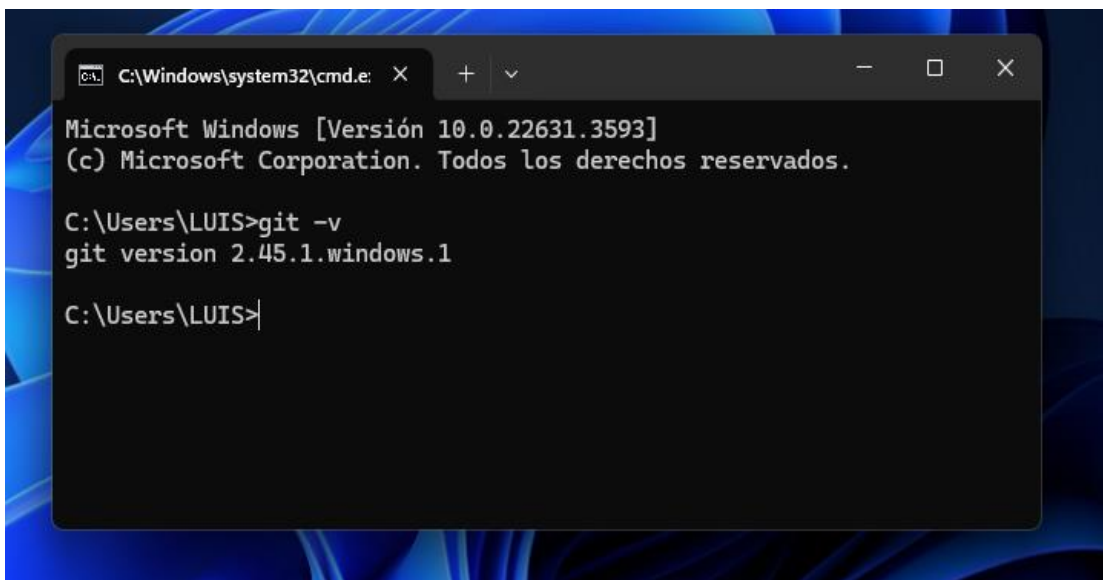
Completar la Instalación

Una vez finalizada la instalación, se verifica abriendo Git Bash y comprobando la versión instalada con el comando ``git --version``.





Comprobando la versión instalada con el comando ``git --version``.



```
C:\Windows\system32\cmd.e: X + v
Microsoft Windows [Versión 10.0.22631.3593]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\LUIS>git -v
git version 2.45.1.windows.1

C:\Users\LUIS>
```

Git Bash

Git Bash es una aplicación para entornos de Windows que proporciona una interfaz de línea de comandos que emula un entorno de bash. Se utiliza principalmente para interactuar con Git, un sistema de control de versiones distribuido, y ejecutar comandos de Unix en Windows. A continuación, detallo algunos usos comunes de Git Bash:

1. Se crea la carpeta donde van a estar todos tus proyecto a cargar en el repositorio y se copia la dirección de la carpeta y se pega en la terminal
C:\Users\LUIS\Desktop\Estudios\Proyectos
2. ***Git init*** se utiliza para crear un nuevo repositorio de Git. Este comando inicializa un nuevo repositorio en el directorio actual, lo que permite a Git comenzar a rastrear los cambios en los archivos de ese directorio

cd ruta/al/directorio/del/proyecto

Utiliza el comando `cd` para moverte al directorio donde quieres inicializar el repositorio de Git.

git init

Ejecuta el comando `git init` para inicializar un nuevo repositorio en el directorio actual



Git Bash

A screenshot of a Git Bash terminal window. The window title is "MINGW64:/c/Users/LUIS". The terminal shows the following commands and output:

```
LUIS@DESKTOP-8GRP7UB MINGW64 ~  
$ cd C:\Users\LUIS\Desktop\Estudios\Proyectos~  
bash: cd: C:\Users\LUIS\Desktop\Estudios\Proyectos~: No such file or directory  
  
LUIS@DESKTOP-8GRP7UB MINGW64 ~  
$ git init  
Initialized empty Git repository in C:/Users/LUIS/.git/  
  
LUIS@DESKTOP-8GRP7UB MINGW64 ~ (master)  
$
```

Two red boxes with numbers are overlaid on the terminal: a box with the number "1" is positioned over the error message "No such file or directory", and a box with the number "2" is positioned over the "git init" command and its output.



CONCLUSIÓN

En el desarrollo de software moderno, Git y GitHub desempeñan roles complementarios pero distintos. Git es un poderoso sistema de control de versiones distribuido que permite a los desarrolladores rastrear y gestionar cambios en su código de manera eficiente a nivel local. Su capacidad para manejar ramas y fusionar cambios facilita el desarrollo paralelo y la colaboración dentro de un equipo.

Git se centra en la gestión local del código, GitHub extiende estas capacidades al ámbito colaborativo y remoto, creando un entorno propicio para el desarrollo de software en equipo. Entender y utilizar ambas herramientas eficazmente es esencial para cualquier desarrollador que busque mejorar la calidad y la eficiencia de su trabajo en proyectos de software