MIND HUB.

Git

Git es un **sistema de control de versiones** distribuido de código abierto y gratuito diseñado para manejar todo, desde proyectos pequeños a muy grandes, con velocidad y eficiencia.
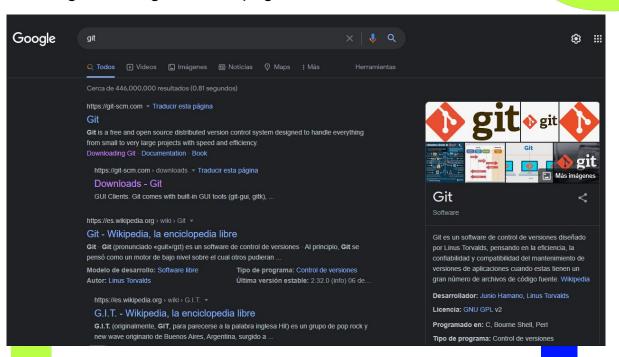
Git es fácil de aprender y ocupa poco espacio con un rendimiento increíblemente rápido . Supera a las herramientas SCM como Subversion, CVS, Perforce y ClearCase con características como bifurcaciones locales económicas , áreas de preparación convenientes y **múltiples flujos de trabajo .**
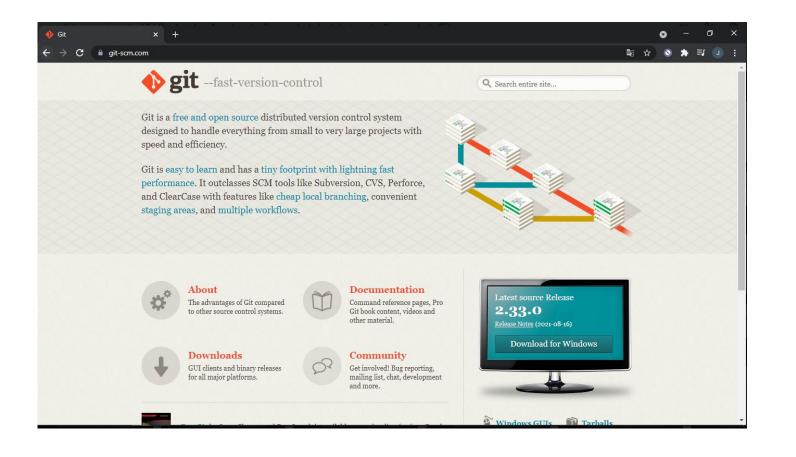
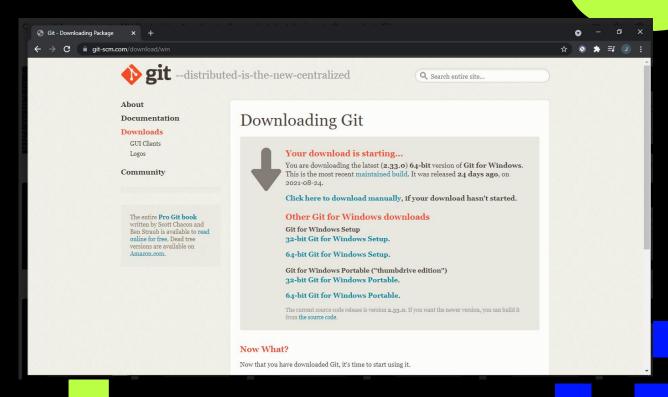# Paso a paso para instalar GIT

PASO1 :

Busca git en tu navegador e ingresa a su página oficial

Paso 2:

Descarga el instalable de Git que corresponda a tu sistema operativo.

Paso 3:

Ejecutar el instalable de git y seleccionar las opciones que verá a continuación.

Git 2.33.0.2 Setup

Choosing the default editor used by Git
Which editor would you like Git to use?

Use Visual Studio Code as Git's default editor

Visual Studio Code is an Open Source, lightweight and powerful editor running as a desktop application. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C#, Java, Python, PHP, Go) and runtimes (such as .NET and Unity).
(WARNING!) This will be installed only for this user.

Use this option to let Git use Visual Studio Code as its default editor.

https://gitforwindows.org/

☐ Only show new options   Back   Next   Cancel

Git 2.33.0.2 Setup

Adjusting the name of the initial branch in new repositories
What would you like Git to name the initial branch after "git init"?

◉ **Let Git decide**
Let Git use its default branch name (currently: "master") for the initial branch in newly created repositories. The Git project intends to change this default to a more inclusive name in the near future.

○ **Override the default branch name for new repositories**
NEW! Many teams already renamed their default branches; common choices are "main", "trunk" and "development". Specify the name "git init" should use for the initial branch:

main

This setting does not affect existing repositories.

https://gitforwindows.org/

☐ Only show new options   Back   Next   Cancel

Git 2.33.0.2 Setup

Adjusting your PATH environment
How would you like to use Git from the command line?

○ **Use Git from Git Bash only**

This is the most cautious choice as your PATH will not be modified at all. You will only be able to use the Git command line tools from Git Bash.

◉ **Git from the command line and also from 3rd-party software**

(Recommended) This option adds only some minimal Git wrappers to your PATH to avoid cluttering your environment with optional Unix tools. You will be able to use Git from Git Bash, the Command Prompt and the Windows PowerShell as well as any third-party software looking for Git in PATH.

○ **Use Git and optional Unix tools from the Command Prompt**

Both Git and the optional Unix tools will be added to your PATH. Warning: This will override Windows tools like "find" and "sort". Only use this option if you understand the implications.

https://gitforwindows.org/

☐ Only show new options   Back   Next   Cancel

Git 2.33.0.2 Setup

Choosing the SSH executable
Which Secure Shell client program would you like Git to use?

◉ **Use bundled OpenSSH**

This uses ssh.exe that comes with Git.

○ **Use external OpenSSH**

NEW! This uses an external ssh.exe. Git will not install its own OpenSSH (and related) binaries but use them as found on the PATH.

https://gitforwindows.org/

☐ Only show new options   Back   Next   Cancel

Git 2.33.0.2 Setup

Choosing HTTPS transport backend
Which SSL/TLS library would you like Git to use for HTTPS connections?

◉ **Use the OpenSSL library**

Server certificates will be validated using the ca-bundle.crt file.

○ **Use the native Windows Secure Channel library**

Server certificates will be validated using Windows Certificate Stores.
This option also allows you to use your company's internal Root CA certificates distributed e.g. via Active Directory Domain Services.

https://gitforwindows.org/

☐ Only show new options   Back   Next   Cancel

Git 2.33.0.2 Setup

Configuring the line ending conversions
How should Git treat line endings in text files?

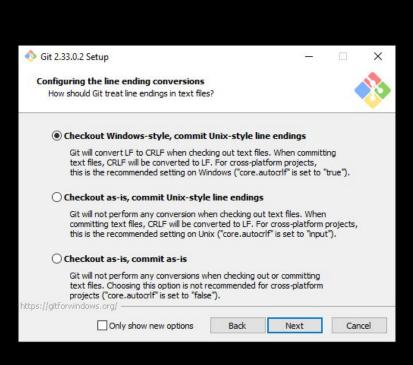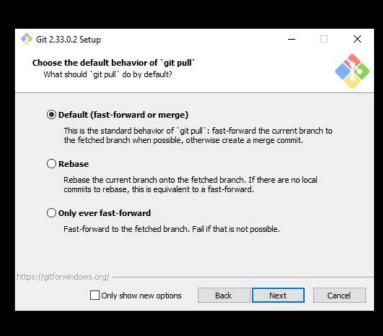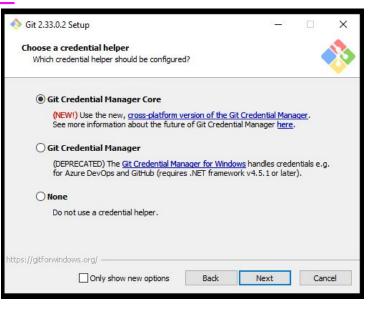◉ **Checkout Windows-style, commit Unix-style line endings**

Git will convert LF to CRLF when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Windows ("core.autocrlf" is set to "true").

○ **Checkout as-is, commit Unix-style line endings**

Git will not perform any conversion when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Unix ("core.autocrlf" is set to "input").

○ **Checkout as-is, commit as-is**

Git will not perform any conversions when checking out or committing text files. Choosing this option is not recommended for cross-platform projects ("core.autocrlf" is set to "false").

https://gitforwindows.org/

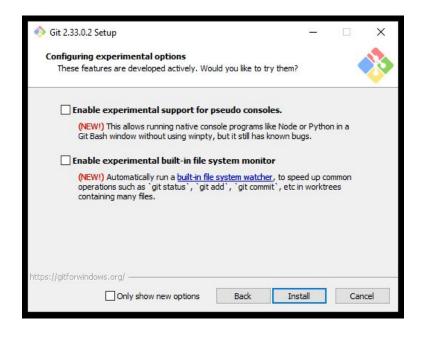☐ Only show new options   Back   Next   Cancel

Git 2.33.0.2 Setup

Configuring the terminal emulator to use with Git Bash
Which terminal emulator do you want to use with your Git Bash?

○ **Use MinTTY (the default terminal of MSYS2)**

Git Bash will use MinTTY as terminal emulator, which sports a resizable window, non-rectangular selections and a Unicode font. Windows console programs (such as interactive Python) must be launched via `winpty` to work in MinTTY.

○ **Use Windows' default console window**

Git will use the default console window of Windows ("cmd.exe"), which works well with Win32 console programs such as interactive Python or node.js, but has a very limited default scroll-back, needs to be configured to use a Unicode font in order to display non-ASCII characters correctly, and prior to Windows 10 its window was not freely resizable and it only allowed rectangular text selections.

https://gitforwindows.org/

☐ Only show new options   Back   Next   Cancel

Git 2.33.0.2 Setup

Choose the default behavior of `git pull`
What should `git pull` do by default?

◉ **Default (fast-forward or merge)**

This is the standard behavior of `git pull`: fast-forward the current branch to the fetched branch when possible, otherwise create a merge commit.

○ **Rebase**

Rebase the current branch onto the fetched branch. If there are no local commits to rebase, this is equivalent to a fast-forward.

○ **Only ever fast-forward**

Fast-forward to the fetched branch. Fail if that is not possible.

https://gitforwindows.org/

☐ Only show new options   Back   Next   Cancel

Git 2.33.0.2 Setup

Choose a credential helper
Which credential helper should be configured?

◉ **Git Credential Manager Core**

(NEW!) Use the new, cross-platform version of the Git Credential Manager.
See more information about the future of Git Credential Manager here.

○ **Git Credential Manager**

(DEPRECATED) The Git Credential Manager for Windows handles credentials e.g.
for Azure DevOps and GitHub (requires .NET framework v4.5.1 or later).

○ **None**

Do not use a credential helper.

https://gitforwindows.org/

☐ Only show new options   Back   Next   Cancel

Git 2.33.0.2 Setup

Configuring extra options
Which features would you like to enable?

☑ **Enable file system caching**

File system data will be read in bulk and cached in memory for certain
operations ("core.fscache" is set to "true"). This provides a significant
performance boost.

☐ **Enable symbolic links**

Enable symbolic links (requires the SeCreateSymbolicLink permission).
Please note that existing repositories are unaffected by this setting.

https://gitforwindows.org/

☐ Only show new options   Back   Next   Cancel
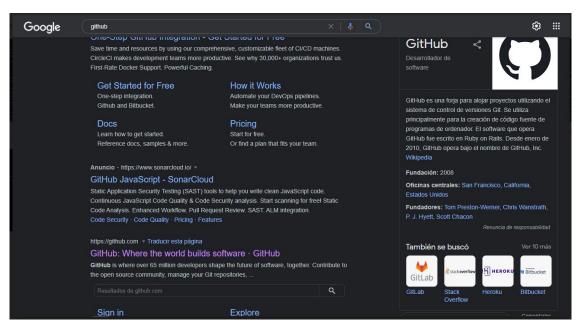
Por último da click en "install"

GITHUB

Esta herramienta los ayudará a crear repositorios remotos, a desplegarlos, y en caso de perder la información de su repositorio local, a tener un backup del mismo, entre otras cosas.

Paso a paso para instalar Github:

Paso 1: Crear una cuenta en github desde el siguiente enlace https://github.com/ o si ya tenes cuenta podes usar la cuenta existente.

Where the world builds software

Millions of developers and companies build, ship, and maintain their software on GitHub—the largest and most advanced development platform in the world.

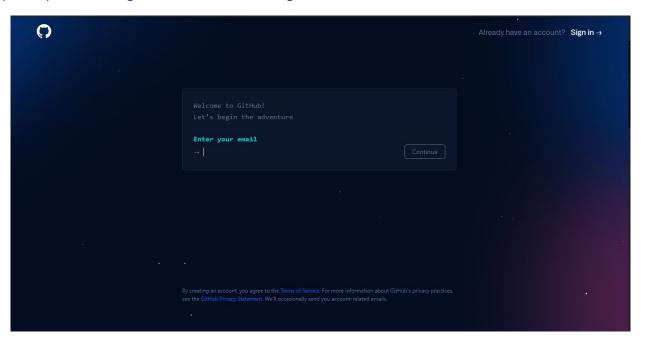Email address   **Sign up for GitHub**

Recorda hacer la siguiente **configuracion** en tu consola de windows :

git config --global user.name "Tu nombre"

git config --global user.email "Tu correo"

para que se configure con tu cuenta de github.

Welcome to GitHub!
Let's begin the adventure

Enter your email

✓ valid@email.com

Create a password

→ ••••••••••••

Continue

━━━━ ━━━━ ━━━━

Password is strong

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter.

By creating an account, you agree to the Terms of Service. For more information about GitHub's privacy practices, see the GitHub Privacy Statement. We'll occasionally send you account-related emails.

Welcome to GitHub!
Let's begin the adventure

Enter your email
✓ valid@email.com

Create a password
✓ password132S

Enter a username
→ ValidUserName132

Continue

ValidUserName132 is available.

Enter a username

✓ ValidUserName132

Would you like to receive product updates and announcements via email?
Type "y" for yes or "n" for no

✓ n

Verify your account

Elija la galaxia en espiral

You're almost done!
We sent a launch code to valid@email.com

→ Enter code

Didn't get your email? Resend the code or update your email address.

Pull requests   Issues   Marketplace   Explore

ValidUserName132
Your personal account

Go to your personal profile

Account settings

Profile

Account

Appearance

Account security

Billing & plans

Security log

Security & analysis

Sponsorship log

Emails ⚠

Notifications

Scheduled reminders

SSH and GPG keys

Repositories

Emails

Please verify an email address to collaborate on GitHub, receive notifications, and help us protect your account.

valid@email.com — Primary ⓘ

- **Unverified** ⓘ Resend verification email
- Not visible in emails ⓘ
- Receives notifications ⓘ

Add email address

[Email address]   Add

Primary email address

Because you have email privacy enabled, valid@email.com will be used for account-related notifications as well as password resets. 90918440+ValidUserName132@users.noreply.github.com will be used for web-based Git operations, e.g., edits and merges.

valid@email.com ⇕   Save

Backup email address

Your backup GitHub email address will be used as an additional destination for security-relevant account notifications

¡Muchas gracias!

MIND HUB.