

Formacion Integral

Carrera Programador full-stack

Git

¿Cómo hace un trabajador para recordar lo que hizo anteriormente?

Discusión:

Un empleado de limpieza necesita recordar cuántas horas trabajó.

Un peluquero necesita recordar cuántos cortes hizo esta semana.

Un carpintero necesita recordar los trabajos que hizo el mes pasado.



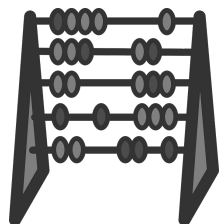
Tal vez basta con una Check list

Registro

Registrar es dejar constancia.

Asentar algo de alguna manera que se pueda recuperar más adelante en el tiempo.

Un registro básico de cantidad podría ser:



¿Y si necesita recordar cuándo y cómo lo hizo ?

Discusión:

Un empleado de limpieza necesita recordar cuándo limpió los estantes altos de la biblioteca.

Un peluquero necesita recordar cuándo atendió a cierto cliente y qué tipo de corte le hizo.

Un carpintero necesita recordar cómo hizo cierta vitrina de copas.



Registro detallado

Es un registro con mayor detalle. No solo basta con colocar una cantidad, requiere una cualidad, una descripción.

Lo que podíamos anotar rápidamente ahora nos lleva más tiempo.

Sin embargo, nos permite recuperar más información.



¿Qué registrate?

¿Recordas cuándo:

le hiciste el último cambio de aceite al auto? O la correa dentada?
fuiste al dentista? O te arreglas ESA muela?

La última vez que comiste pescado?

Que tu hijo tuvo consulta de rutina con el pediatra? Vacunas?

Cuánto te sacaste en matemáticas en 4to año de secundaria?

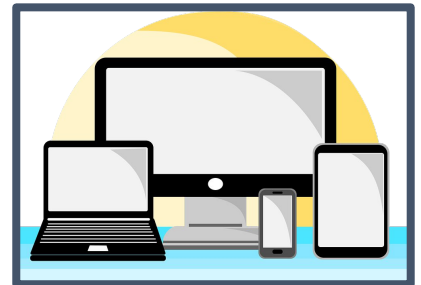
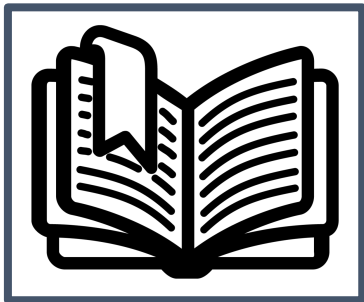
...Y qué contenidos matemáticos “viste”?

Pero... por más ordenado que seas, no podes guardar todos esos registros en papel.

Registro digitalizado

Podría ser lo mismo que un registro en un cuaderno pero en algún medio digital.

¿Qué ventajas podría traer esto?



¿Y si el trabajo es colaborativo?

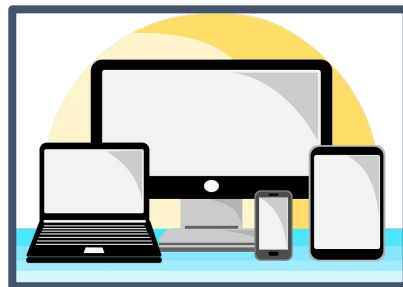
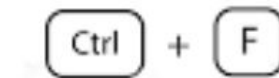
Si te pregunto:

¿Qué día fue tu entrevista de trabajo el año pasado?

¿Qué referencias escribiste en tu último currículum?

¿Dónde está la receta de pollo al disco con la que la rompiste?

¿Cómo lo buscas: Agenda vs Compu?



¿Y si el trabajo es colaborativo?

Supongamos ahora que no trabajas solo.

Tus apuntes en un cuaderno difícilmente permitirán avanzar a tu compañero.

Si guardamos información digital pero no lo compartimos tampoco.

¿Qué alternativas tendríamos?

¿Y si el trabajo es colaborativo?

Basado en lo que conocemos pensemos herramientas para:

- Escribir un documento colaborativo.
- Llevar un registro de ventas y clientes.
- Tareas realizadas y tareas pendientes.
- Reuniones realizadas y pendientes

¿Y si el trabajo es colaborativo?

Tareas (pero... dejemos esto para el final):

En grupos trabajar colaborativamente:

- 1- Registrar las ventas de dos sucursales de un negocio y sumarlas.
- 2- Escribir un resumen de lo visto en esta clase.
- 3- Hacer una presentación donde e ofrezca un servicio de programación.

¿Cómo trabajan en equipo los programadores?

Discusión:

Supongamos que tenemos que realizar una tarea de programación en grupos de a 2. ¿Qué alternativas tenemos para trabajar en equipo?

¿si fuera un proyecto grande en el que participarían 10 personas?

¿Y si por razones de tiempo habría que separar las tareas en subtareas para después unir todo junto más tarde?

Git



Git es un sistema de control de versiones distribuido de código abierto desarrollado por Linus Torvalds, el creador de Linux.

Está pensando en la *eficiencia*, la *confiabilidad* y *compatibilidad* del mantenimiento de *versiones* de aplicaciones cuando estas tienen un gran número de archivos de código fuente.

Su propósito es llevar *registro* de los *cambios* en archivos de computadora incluyendo *coordinar* el *trabajo* que varias personas realizan sobre archivos compartidos en un *repositorio* de código.

Gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración del mismo.



Time

Your
Project

VCS

Add headline
to index page

Create "about" page



Change page layout

index.html

modified

```
<h1>Headline</h1>
...
```

about.html

created

```
<html>
<head>
...
```

photo.png

created

about.html

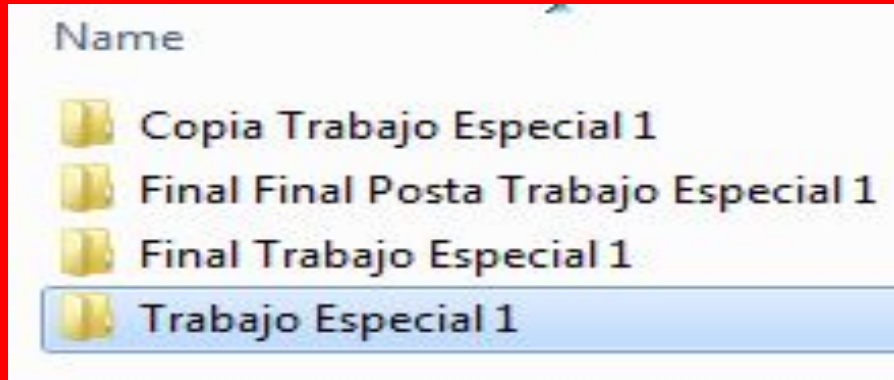
modified

```
<div>new content</div>
...
```



Un Sistema de Versionado de Código (SVC) es lo que nos permite **compartir el código** fuente de nuestros desarrollos y a la vez **mantener un registro de los cambios** por los que va pasando.

Copiar y Pegar Archivos



NO

Es control de versiones

Software de Control de Versiones



TERMINOLOGÍA:

Repositorio: Se almacenan los datos actualizados e históricos de cambios.

Módulo: Conjunto de directorios y/o archivos dentro del repositorio.

Revisión: Es una versión determinada de la información que se gestiona.

Checkout: Un despliegue crea una copia de trabajo local desde el repositorio.

Commit: Sucede cuando una copia de los cambios hechos a una copia local es escrita o integrada sobre el repositorio.

Mas terminología: “conflict”, “resolve”, “diff”, “export”, “import”, “merge”, “sync”, “workspace”.

Balance

- + Se puede seguir trabajando offline. Incluso si se cae el servidor.
- + Cada repositorio tiene toda la información histórica (Backups replicados).
- + Repositorios más limpios ([Dictador Benevolente](#)).
- + Server de Git consume menos recursos.
- + Permite hacer pruebas locales versionadas y subir solo lo relevante.
- Curva de aprendizaje mayor.



Git is a [free and open source](#) distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

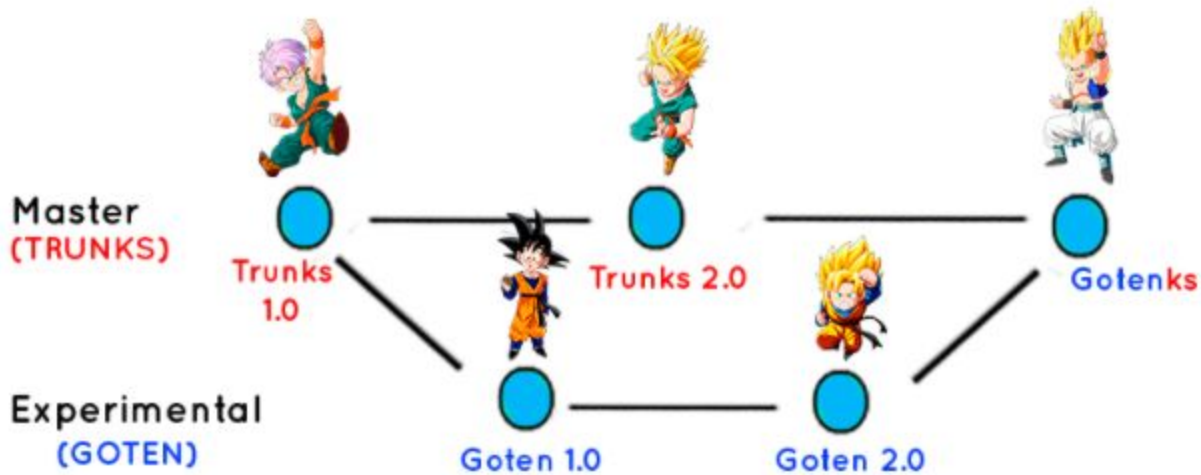
GIT

CONTROL DE VERSIONES



GIT

Es la creación de un nuevo commit que vincule una de nuestras ramas (**trunks**), con alguna otra rama (**Goten**), dando como resultado un nuevo commit (**Gotenks**).

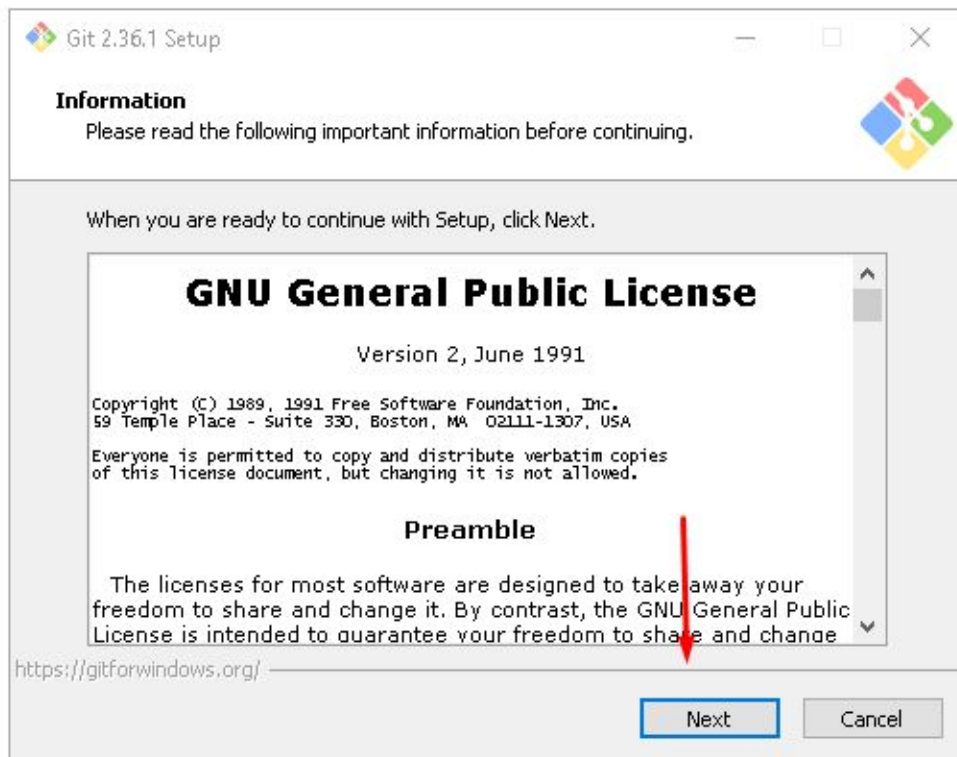


Instalacion de Git

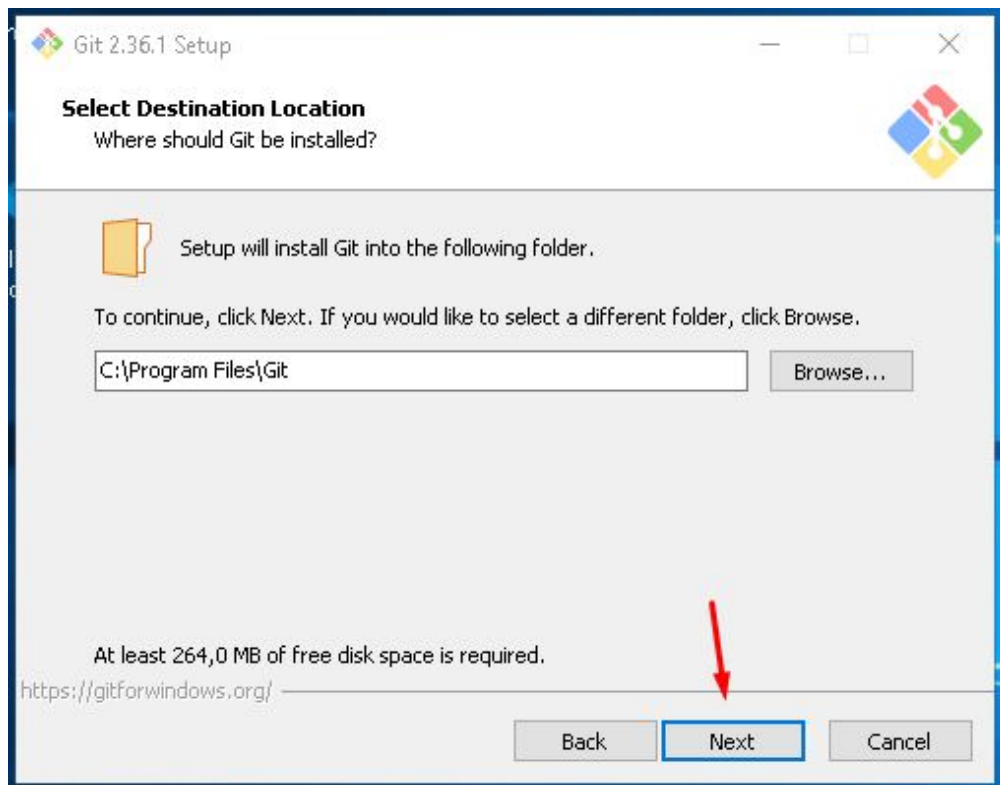
Entramos al siguiente link <https://git-scm.com/> y realizamos la descarga de la versión de Git



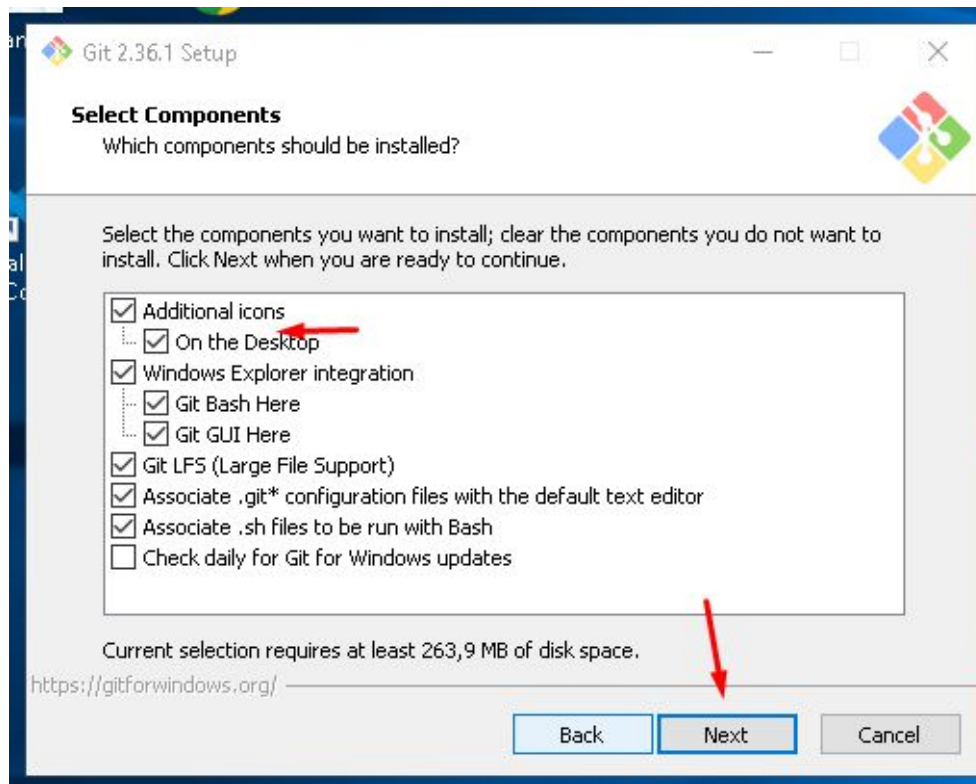
Instalacion de Git



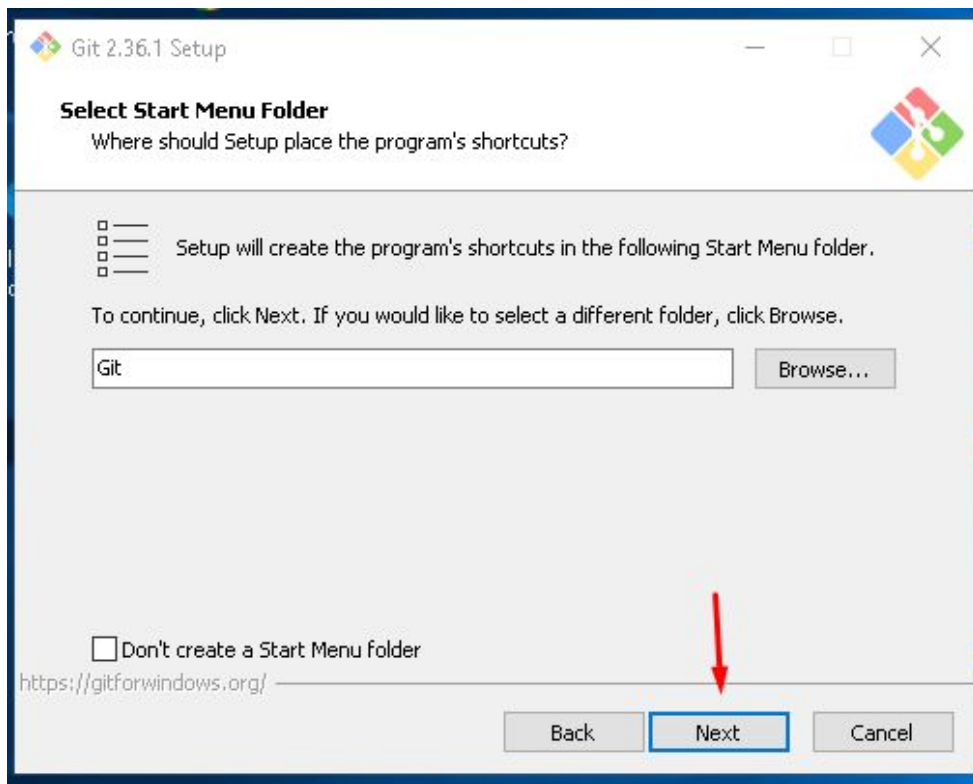
Instalacion de Git



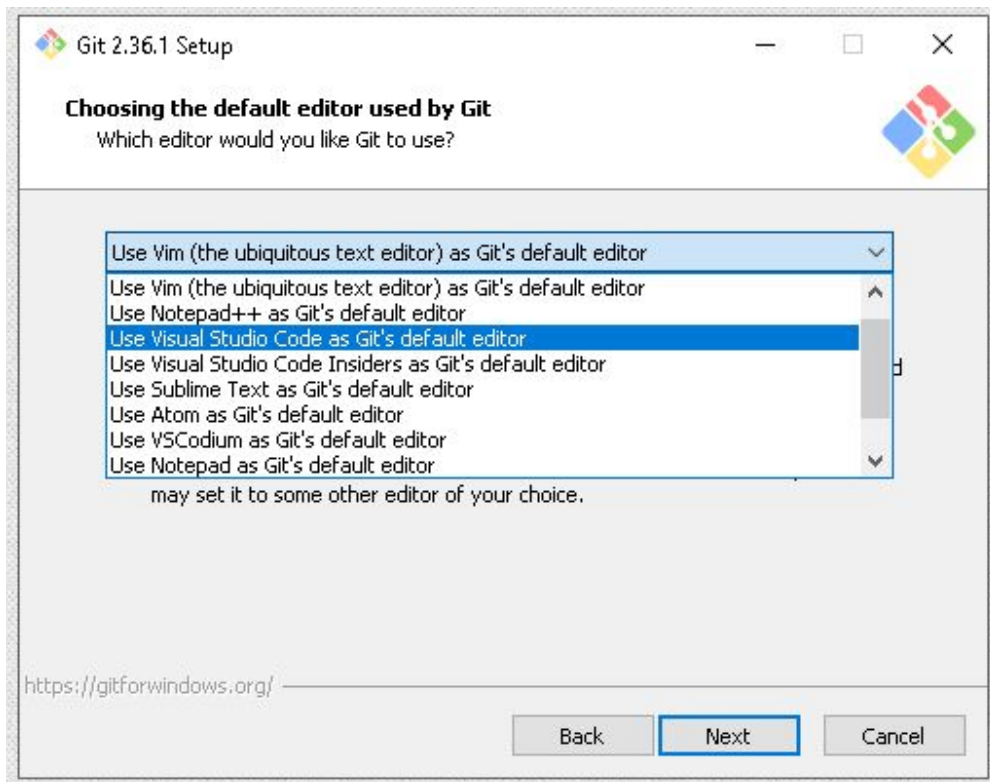
Instalacion de Git



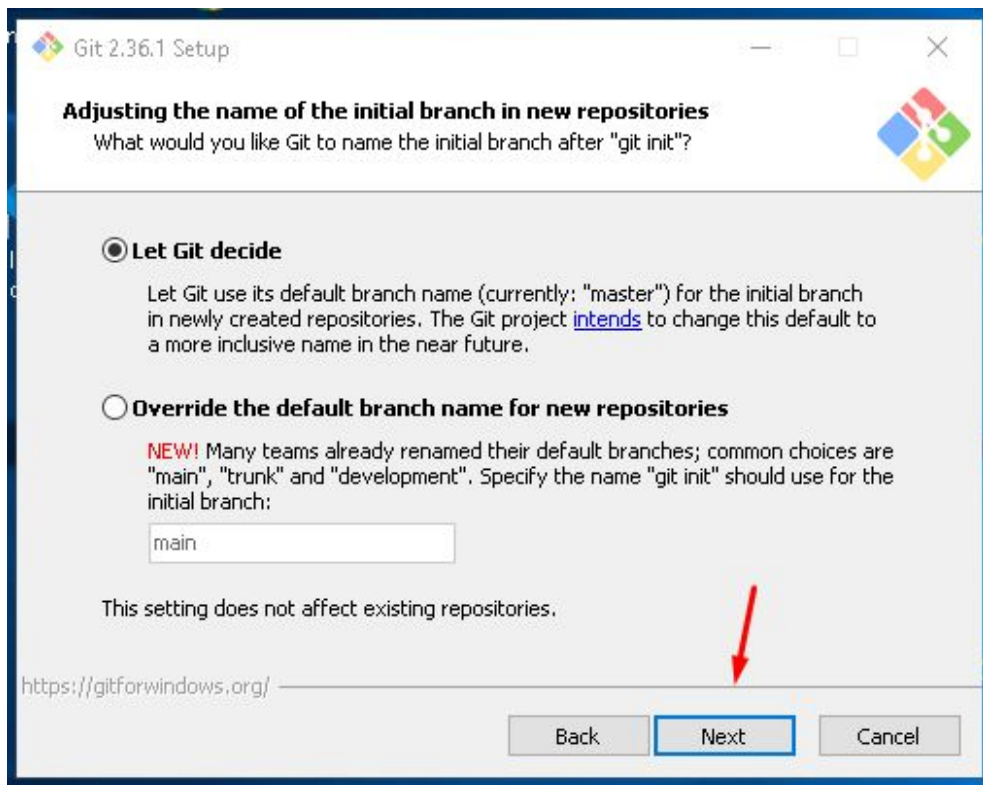
Instalacion de Git



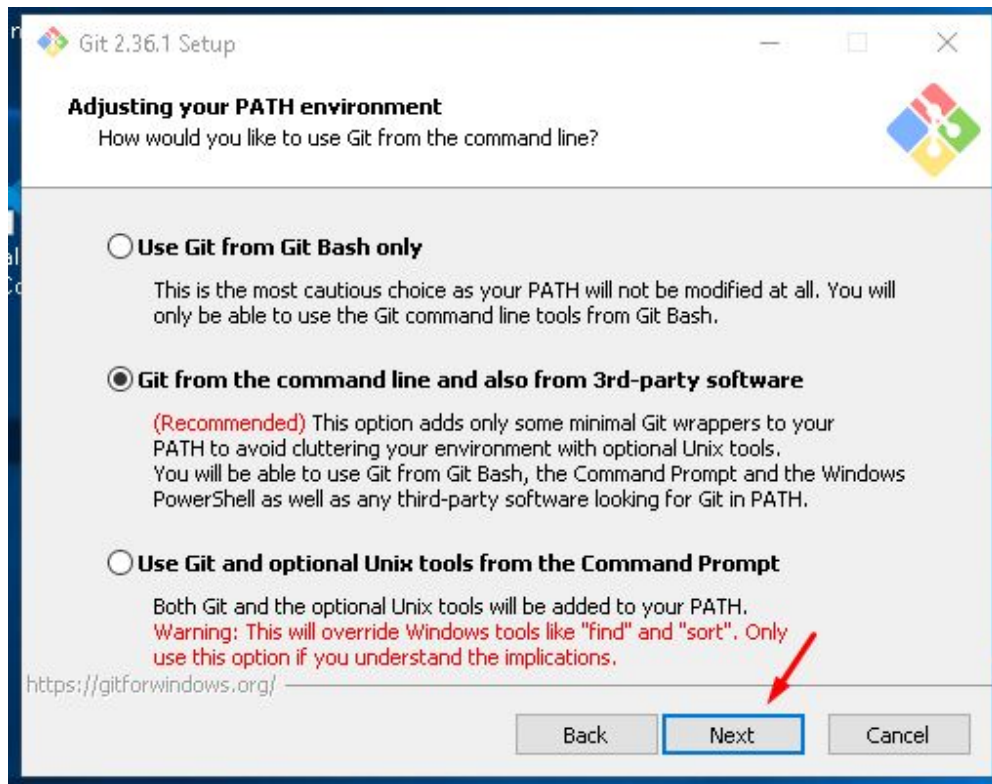
Instalacion de Git



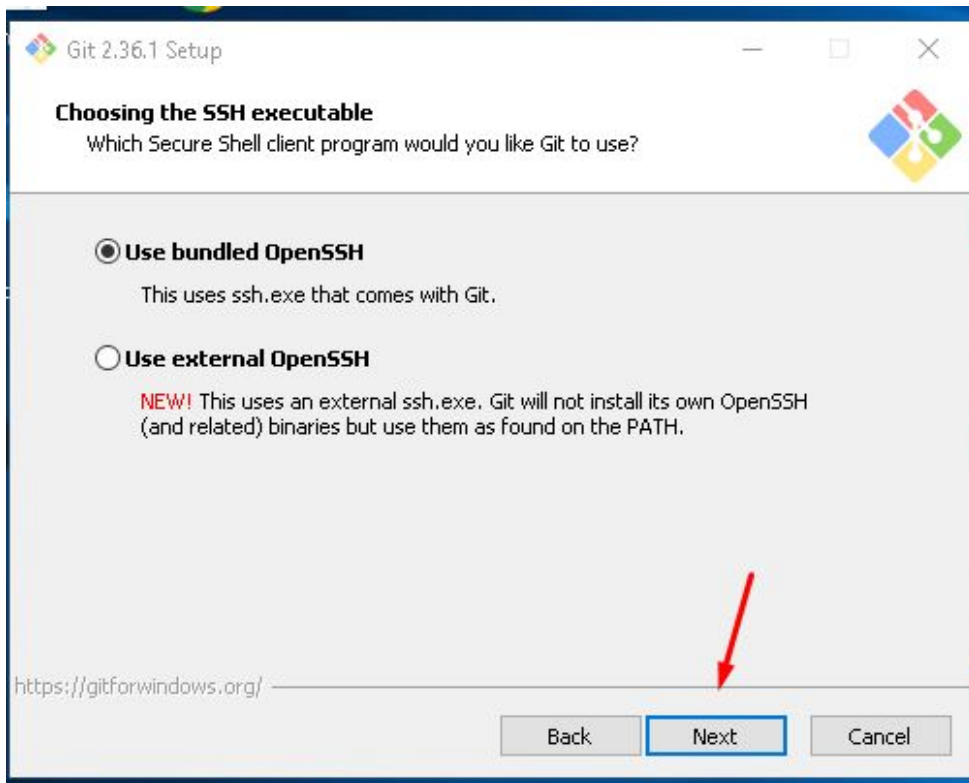
Instalacion de Git



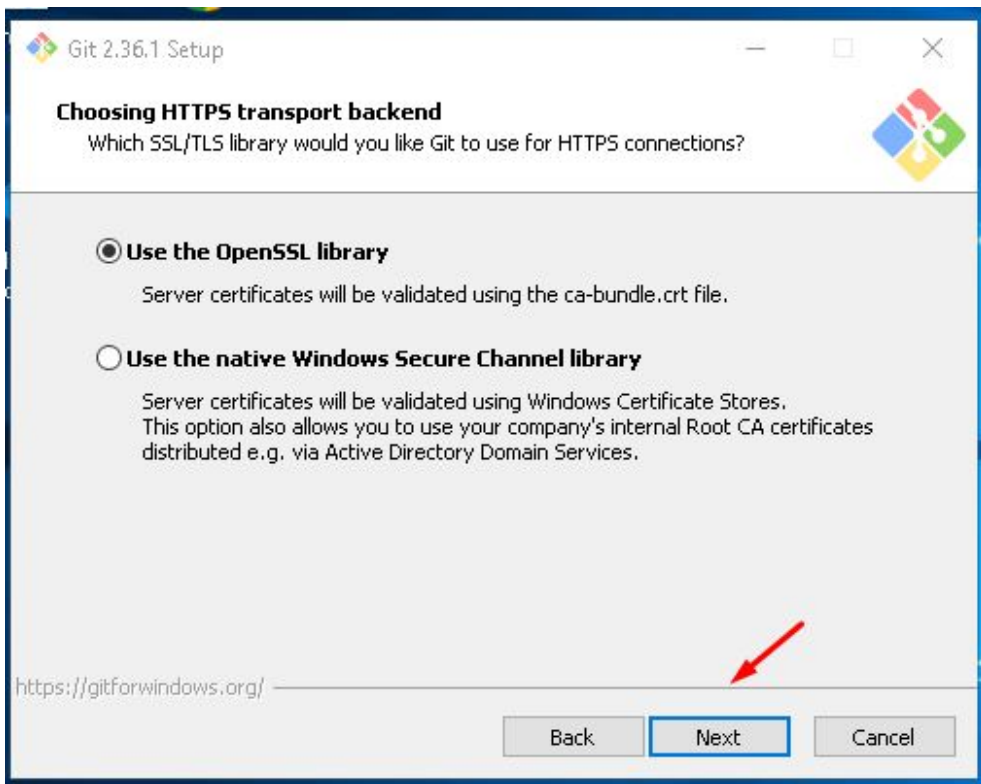
Instalacion de Git



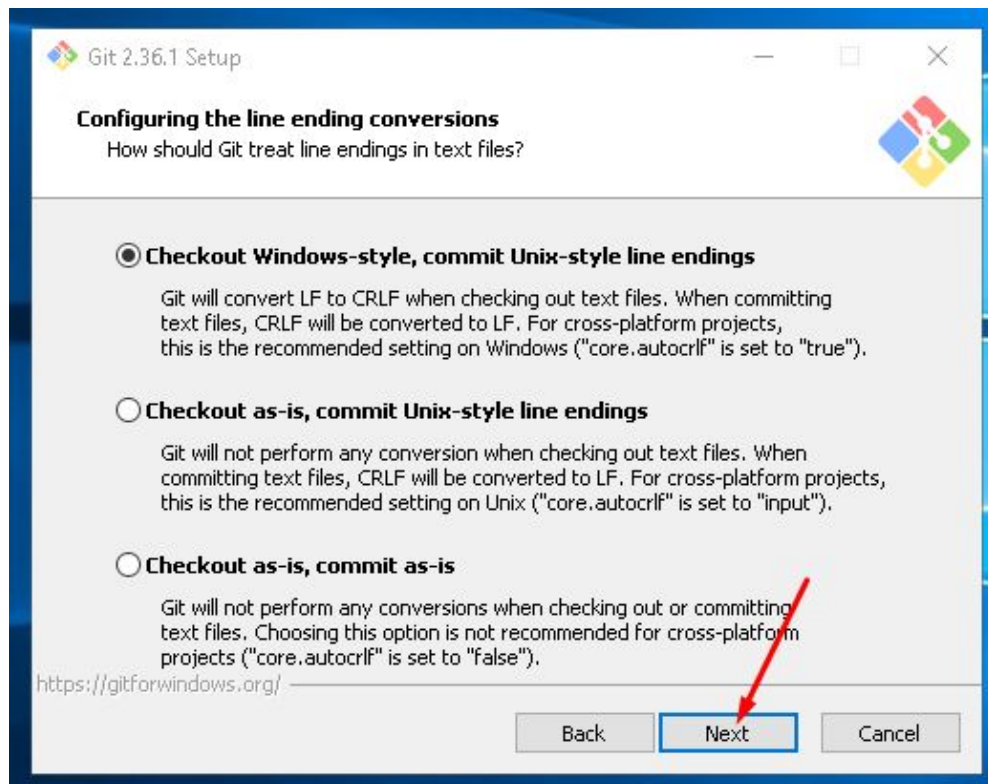
Instalacion de Git



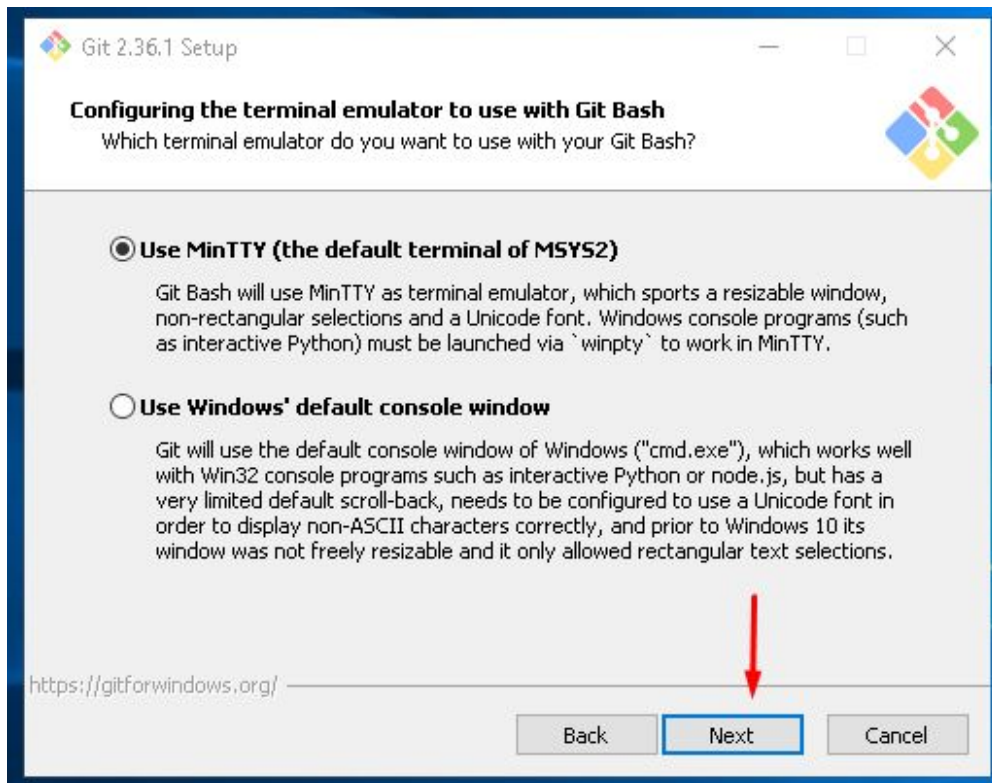
Instalacion de Git



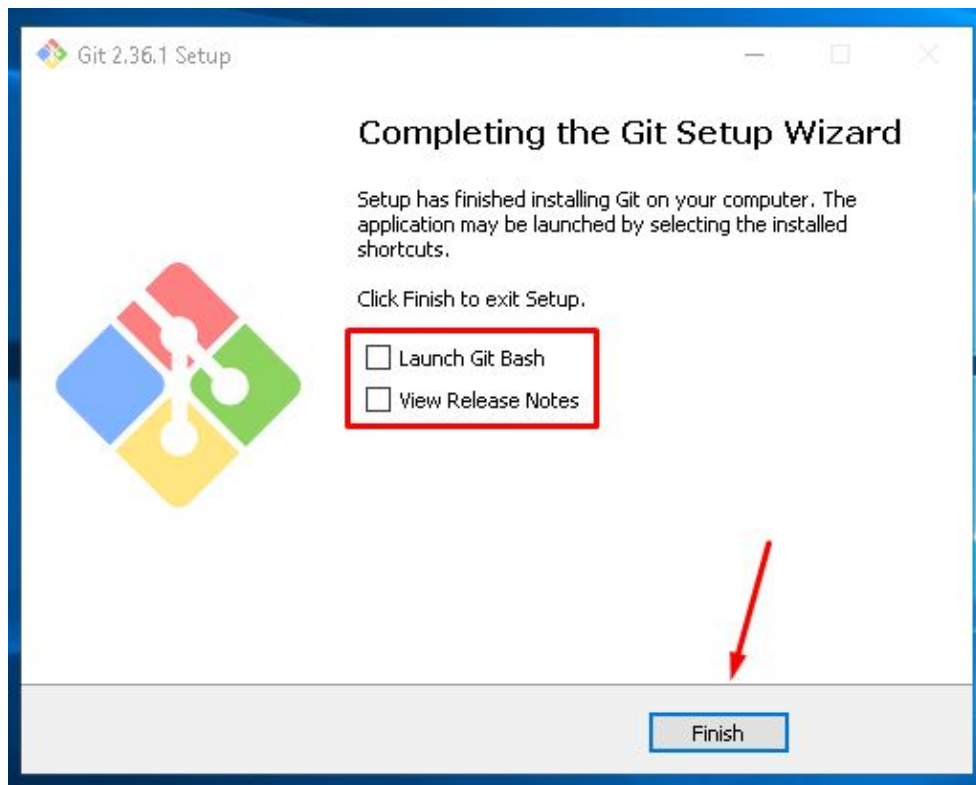
Instalacion de Git



Instalacion de Git



Instalacion de Git



Formación Integral

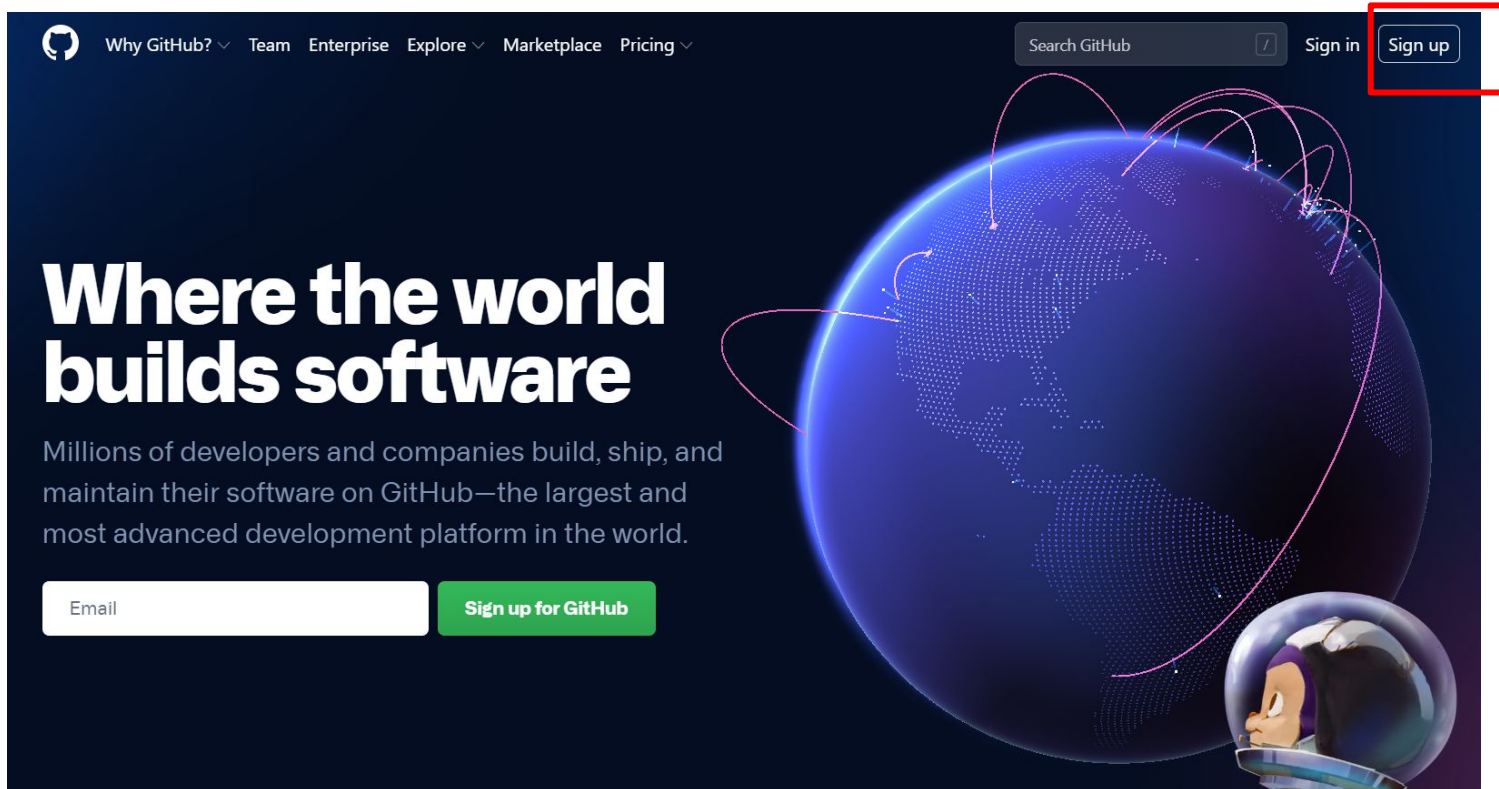
Carrera
Programador
full-stack

GITHUB



- Plataforma de desarrollo colaborativo, que utiliza Git.
- Ofrece GIT y más cosas juntas
- Gratuito
- Tiene facetas de red social. Se suele usar como CV de proyectos propios.
- Existen otras alternativas (GitLab, BitBucket, etc).

Crear una cuenta



Crear una cuenta

Username *

cfiFullstackDocente ✓

Email address *

cfi403pfs.docente.2021@gmail.com ✓

Password *

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)


Email preferences

☐ Send me occasional product updates, announcements, and offers.

Verify your account

Solucione este rompecabezas para que sepamos que es una persona real

Verificar



Create account

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.



Crear una cuenta

Welcome to GitHub

Woohoo! You've joined millions of developers who are doing their best work on GitHub. Tell us what you're interested in. We'll help you get there.

What kind of work do you do, mainly?

Software Engineer

I write code

Student

I go to school

Product Manager

I write specs

UX & Design

I draw interfaces

Data & Analytics

I write queries

Marketing & Sales

I look at charts

Teacher

I educate people

Other

I do my own thing



Crear una cuenta

How much programming experience do you have?

None

I don't program at all

A little

I'm new to programming

A moderate amount

I'm somewhat experienced

A lot

I'm very experienced

What do you plan to use GitHub for?

(Select up to 3)



Learn to code



Learn Git and GitHub



Host a project (repository)



Create a website with
GitHub Pages



Collaborating with my team



Find and contribute to
open source



School work and student
projects



Use the GitHub API



Other



Crear una cuenta

I am interested in:

languages, frameworks, industries

We'll connect you with communities and projects that fit your interests.

For example: `openfaas` `server` `gatsby`

Complete setup



Crear una cuenta



Please verify your email address

Before you can contribute on GitHub, we need you to verify your email address.

An email containing verification instructions was sent to `cfl403pfs.docente.2021@gmail.com`.

[Resend verification email](#)[Change your email settings](#)

Crear una cuenta



Almost done, @cflFullstackDocente!

To complete your GitHub sign up, we just need to verify your email address:
cfl403pfs.docente.2021@gmail.com.

Verify email address



Crear un repositorio

Your email was verified.

What do you want to do first?

Every developer needs to configure their environment, so let's get your GitHub experience optimized for you.



Start a new project

Start a new repository or bring over an existing repository to keep contributing to it.

Create a repository



Collaborate with your team

Improve the way your team works together and get access to more features with an organization.

Create an organization



Learn how to use GitHub

Get started with an "Introduction to GitHub" course in our Learning Lab.

Start Learning



Crear un repositorio

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner *



cflFullstackDocente ▾

Repository name *

/ CFL2021 ✓

Great repository names are short and memorable. Need inspiration? How about [sturdy-octo-dollop?](#)

Description (optional)



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☐ Add a README file

This is where you can write a long description for your project. [Learn more.](#)

☐ Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)

☐ Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

Create repository



Crear un repositorio

cflFullstackDocente / CFL2021

Unwatch 1 Star 0 Fork 0

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

Quick setup — if you've done this kind of thing before

[Set up in Desktop](#) or [HTTPS](#) [SSH](#) 

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).


...or create a new repository on the command line

```
echo "# CFL2021" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/cflFullstackDocente/CFL2021.git
git push -u origin main
```



...or push an existing repository from the command line

```
git remote add origin https://github.com/cflFullstackDocente/CFL2021.git
git branch -M main
git push -u origin main
```



...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

[Import code](#)

 **ProTip!** Use the URL for this page when adding GitHub as a remote.



Formación Integral

Carrera Programador full-stack

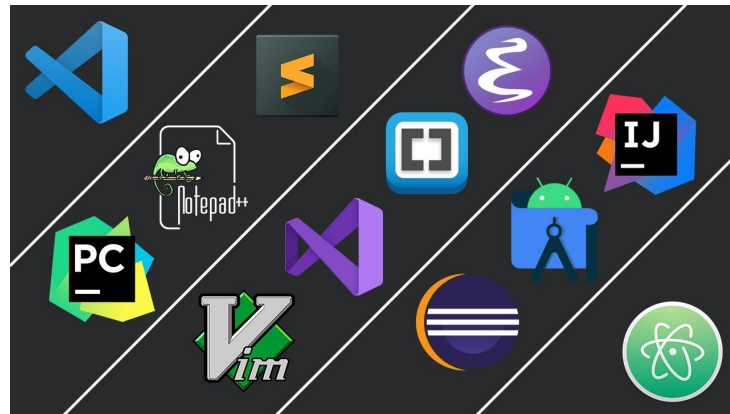
Editor de código

¿Qué es un editor de código?

Un editor de código fuente es un **editor de texto diseñado específicamente para editar el código fuente de programas informáticos**. Puede ser una aplicación individual o estar incluido en un entorno de desarrollo integrado.

Algunos que podemos nombrar son:

- **Sublime Text**
- **Notepad++**
- **Atom**
- **Visual Studio Code**
- **Vim**



Visual Studio Code

¿Qué es Visual Studio Code?

Uno de los programas que no pueden faltar en el repertorio de cualquier programador es un editor de código.

Visual Studio Code es un **editor de código totalmente gratuito y OpenSource** desarrollado por Microsoft.

Visual Studio Code

Ventajas

- **Fácil instalación**
- **Compatible con la mayoría de los lenguajes de programación**
- **VSC está disponible para Windows, macOS y Linux**, por lo que, independientemente del sistema operativo que utilicemos

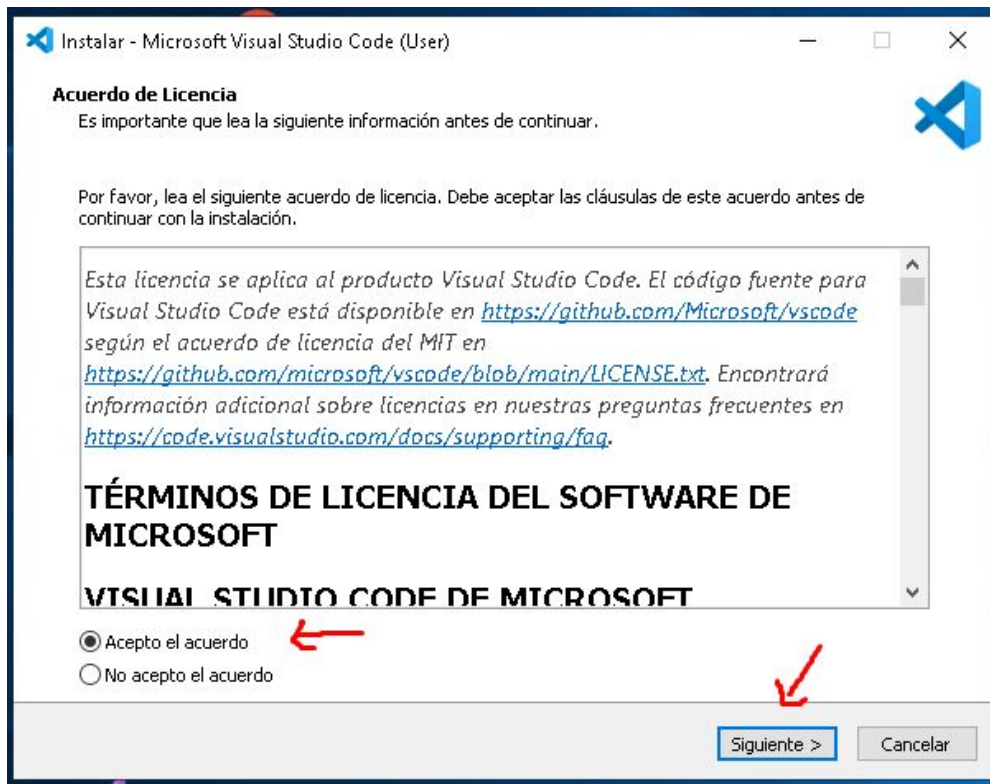
Instalación de visual code

Entramos al siguiente link para descarga

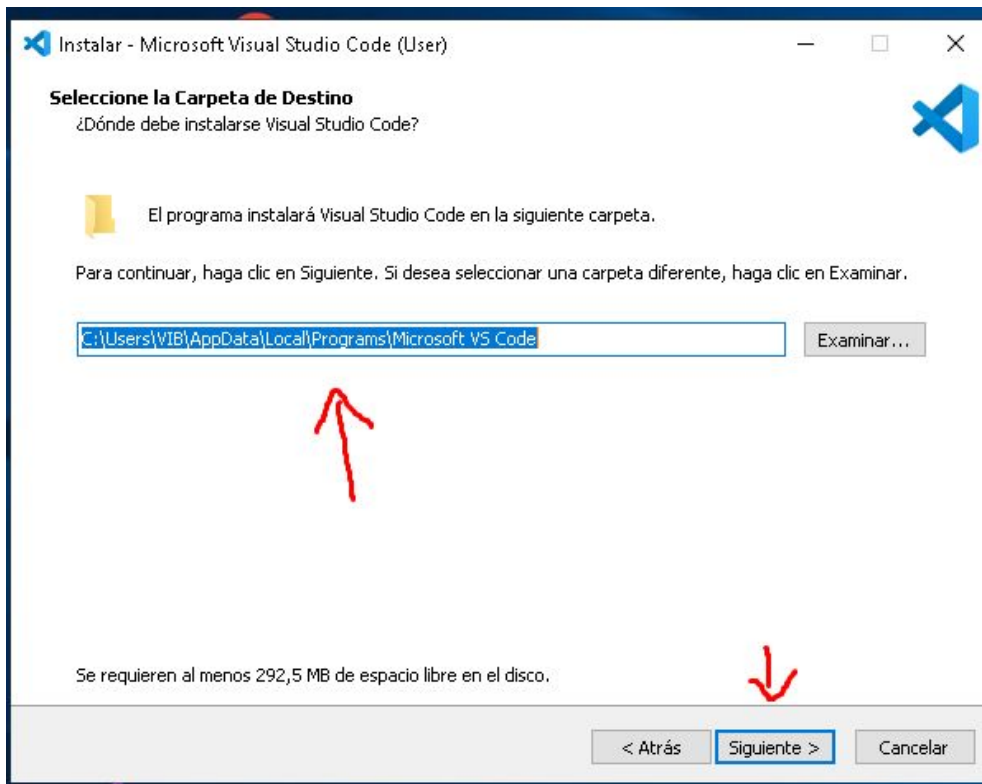
<https://code.visualstudio.com/>

Descargamos el archivo y seguimos los pasos.

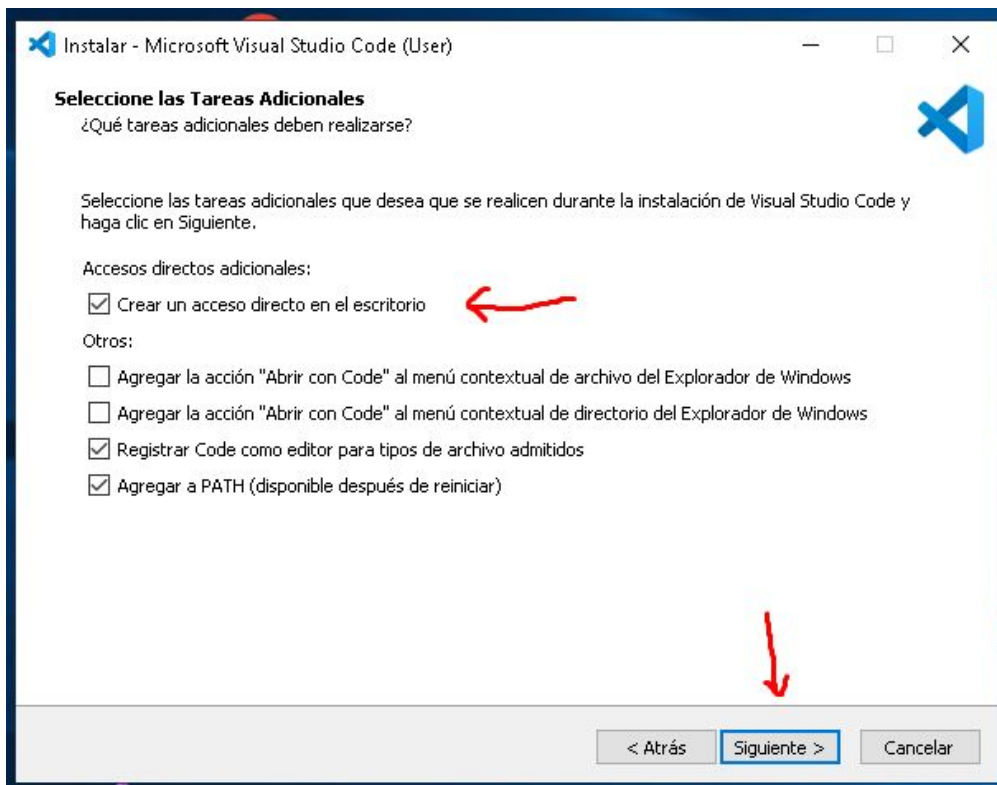
Instalación de visual code



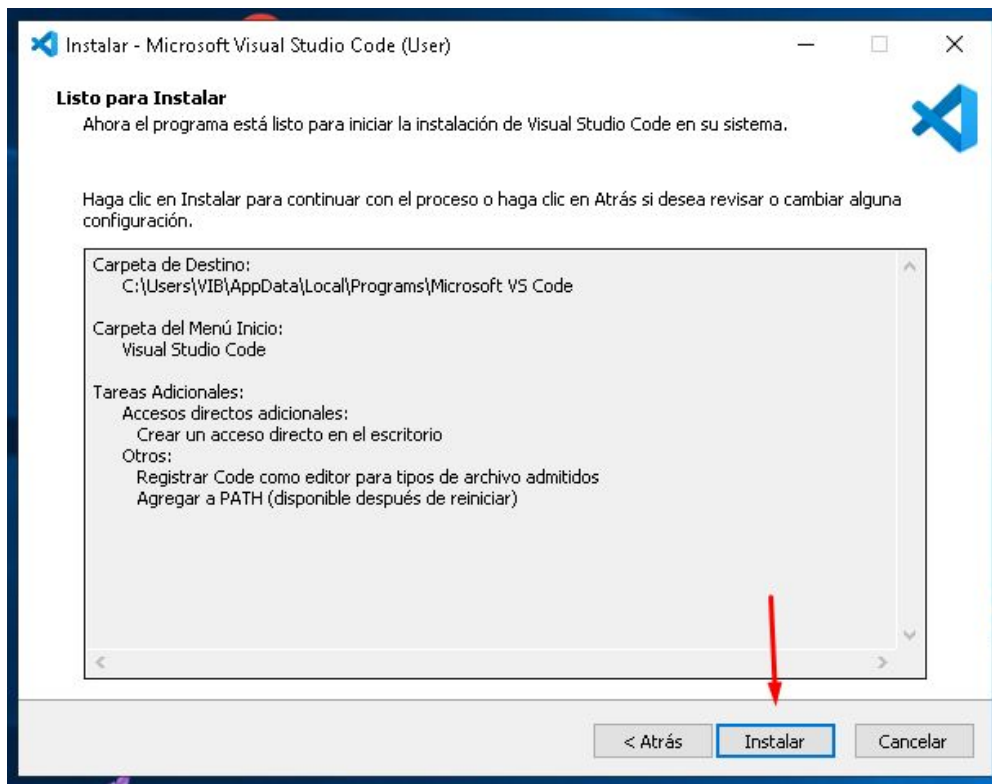
Instalación de visual code



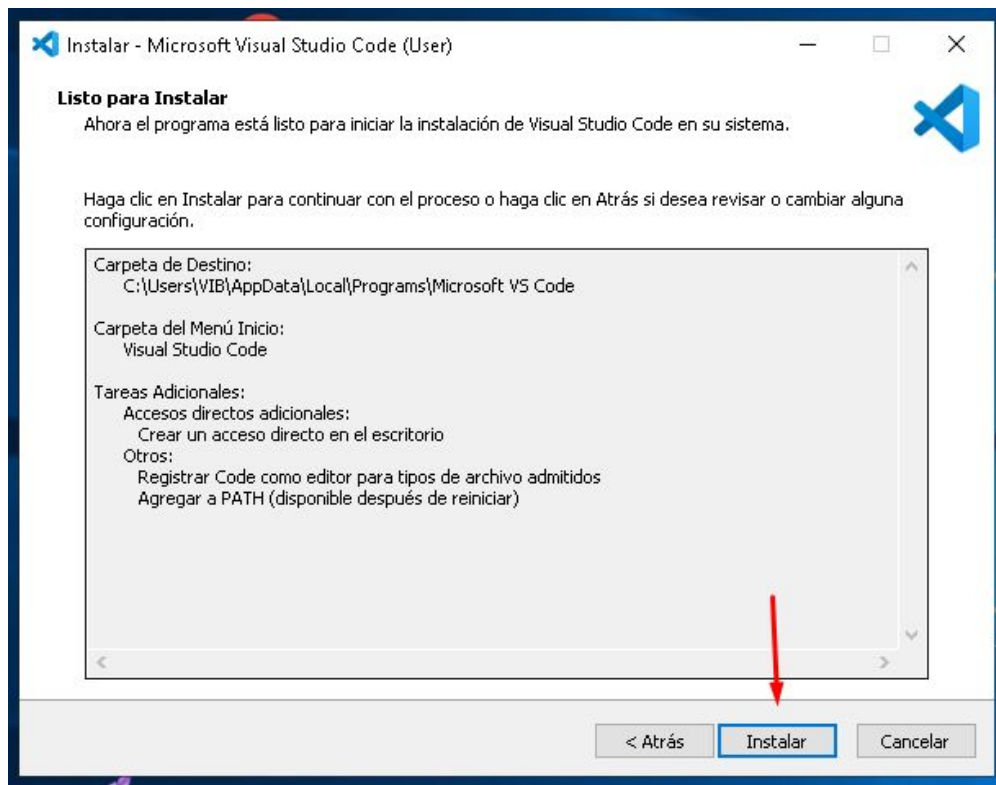
Instalación de visual code



Instalación de visual code



Instalación de visual code



Instalación de visual code

Visual Studio Code

Edición mejorada

Inicio

 Nuevo archivo...

 Abrir archivo...

 Abrir carpeta...

Reciente

No tiene carpetas recientes, [abrir una carpeta](#) iniciar.

Tutoriales



Introducción a VS Code

Descubra las mejores personalizaciones para configurar VS Code a su manera.



Conozca los aspectos básicos

Vaya directamente a VS Code y obtenga información general sobre las características imprescindibles.



Aumente su productividad

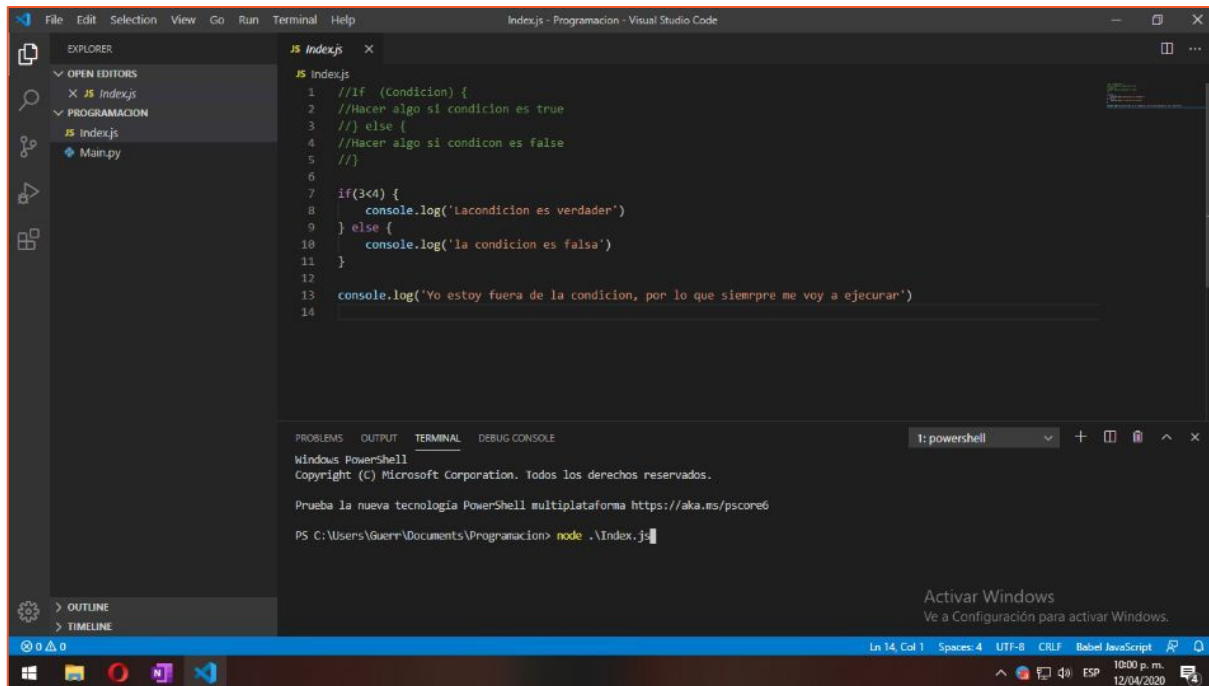
Formacion Integral

Carrera
Programador
full-stack

Comandos de Git

Comandos para utilizar Git

Desde la consola/terminal de VSC(Visual Studio Code)



Comandos para utilizar Git

Paso 1:

El entorno de desarrollo es lo que tenemos en nuestra máquina local. Las tres partes son nuestro directorio de trabajo, el área de ensayo y el repositorio local.

- Elijamos un lugar en el que deseamos colocar nuestro entorno de desarrollo .
- Debemos ser organizad@s para poder luego trabajar de manera ordenada en nuestro repositorio local

Configurar mi usuario como autor

A cada commit se le guarda el autor con un username y mail

```
git config --global user.name mi_nombre
```

```
git config --global user.email mi_correo
```

--global hace que se guarde como configuración del usuario

Si no es solo para cada repositorio

Podes commitear en la misma PC con mail diferente (laboral/personal) en cada repositorio

Paso 2

Ahora queremos tomar un Repositorio Remoto y poner lo que contiene en su máquina.

- Para hacer eso usamos **git clone** https://github.com/my_repo

Como podemos ver en el diagrama a continuación, esto copia el repositorio remoto en dos lugares, tu directorio de trabajo y el repositorio local.

Ahora ves cómo se distribuye el control de versiones de git. El repositorio local es una copia del remoto y actúa como tal. La única diferencia es que no lo compartes con nadie.



Como que el repo clonado en nuestro local

Creamos el repo en GitHub

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?

[Import a repository.](#)

Owner *



FipAyacucho ▾

Repository name *



SedeSanCayetano



Great repository names are short and memorable. Need inspiration? How about [studious-memory](#)?

Description (optional)



Public

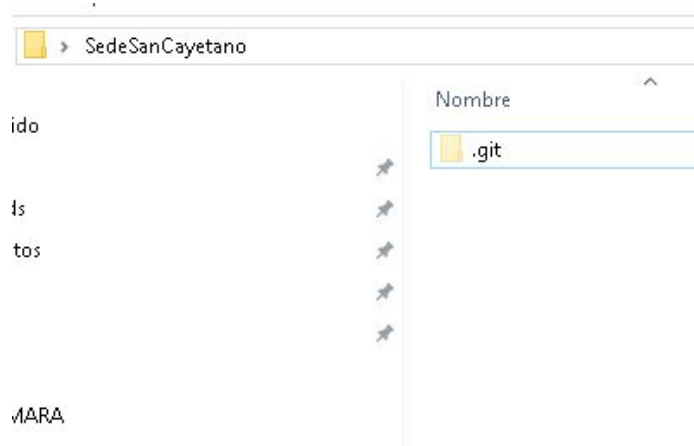
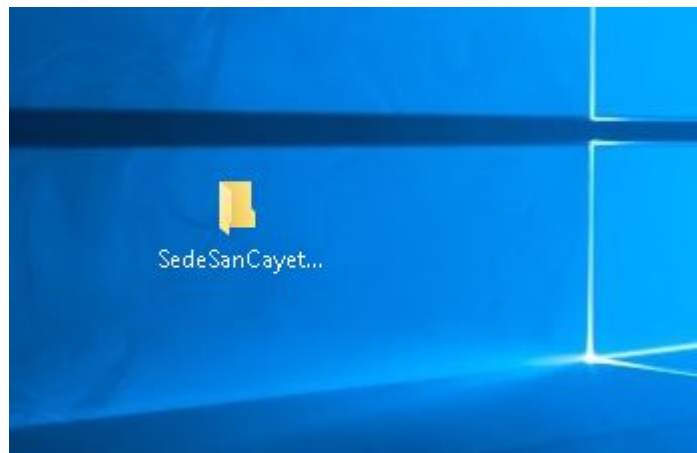
Anyone on the internet can see this repository. You choose who can commit.

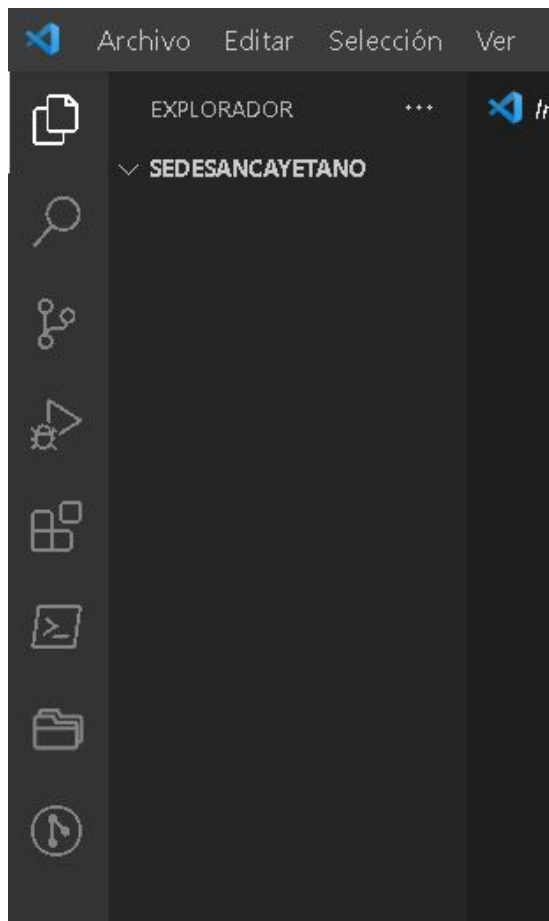


Private

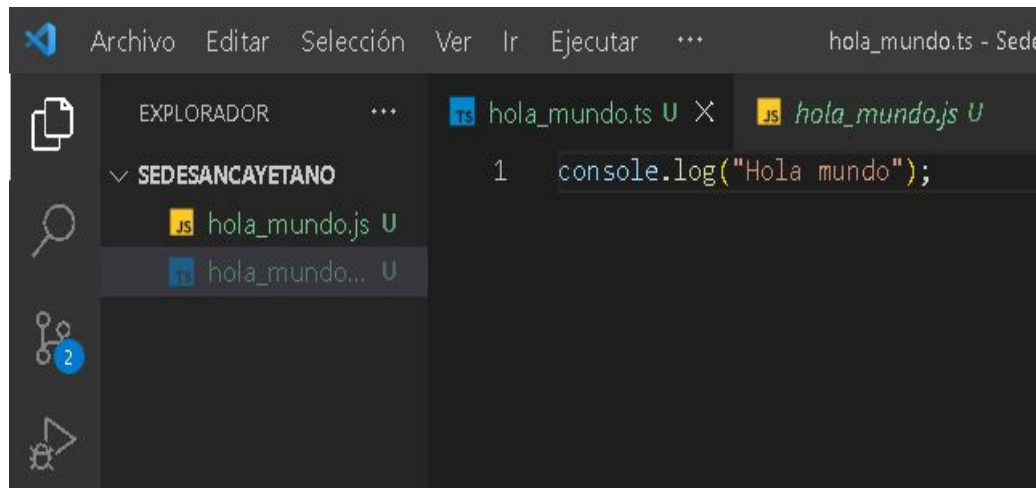
You choose who can see and commit to this repository.

Aqui clonamos el repo de GitHub (repositorio remoto) a nuestro equipo repositorio local
Como se ve se creo una carpeta con el nombre .git que nos indica que es un repositorio





Así se ve el repositorio clonado en Visual Studio Code

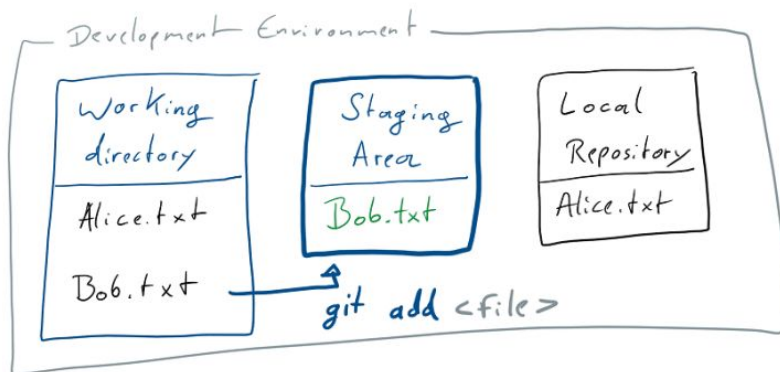


Aquí comenzamos a trabajar dentro de nuestro repositorio

Paso 3:

A continuación creamos un nuevo file y lo vamos a agregar:

- **git status** (para verificar qué cambios tenemos para agregar)
- **git add** tu_file lo agregamos.
- Luego volvemos a hacer **git status** para ver que cambios hubo

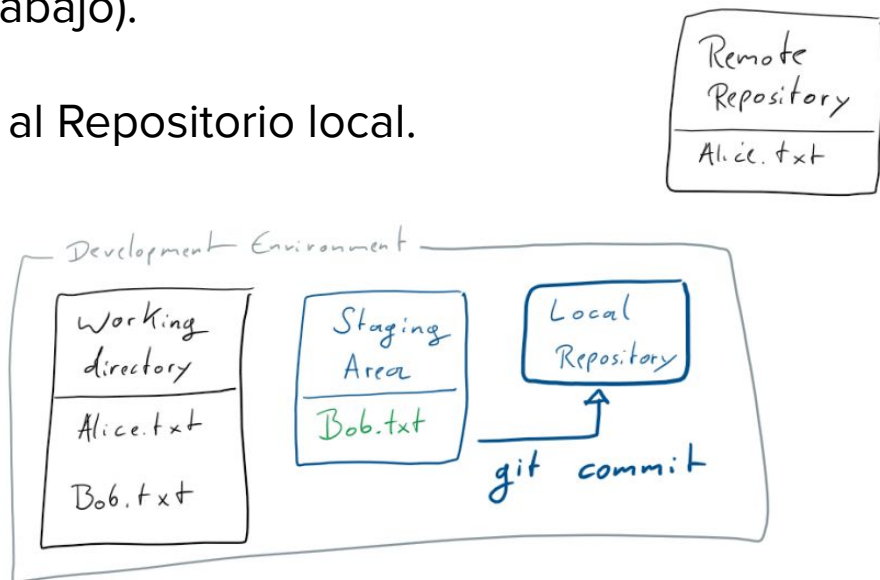


Paso 4:

Cuando hayas agregado todos tus cambios (que en este momento solo es tu file nuevo)

- **git commit -m** “Mensaje”(Los cambios recopilados que commiteas son una parte importante del trabajo).

Nota: Tu commit se agrega al Repositorio local.



El comando **git add** añade un cambio del directorio de trabajo en el entorno de ensayo. De este modo, indica a Git que quieres incluir actualizaciones en un archivo concreto en la próxima confirmación.

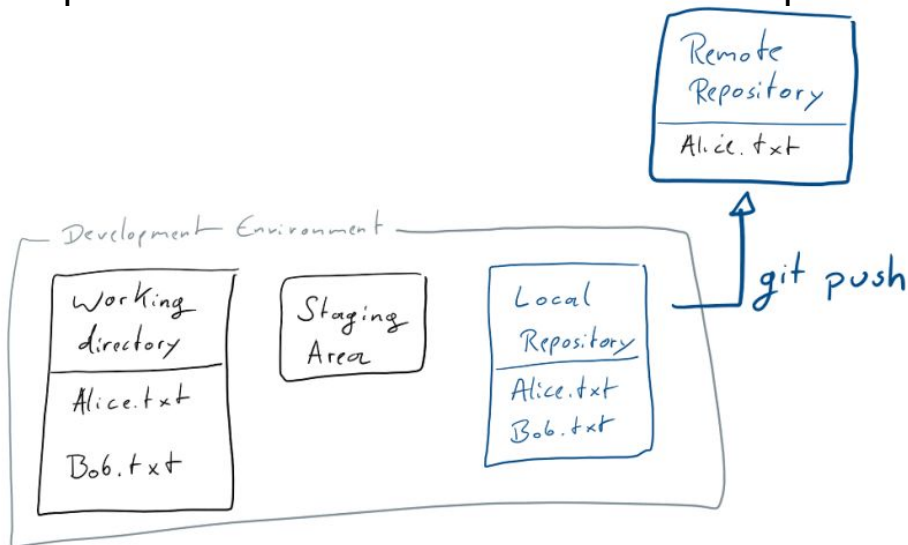


El comando **git commit** captura una instantánea de los cambios preparados en ese momento del proyecto. Las instantáneas confirmadas pueden considerarse como versiones "seguras" de un proyecto: Git no las cambiará nunca a no ser que se lo pidas expresamente.

Paso 5:

Ahora sus cambios están en su repositorio local, que es un buen lugar para estar mientras nadie más los necesite o aún no esté listo para compartirlos.

- **git push** (Para compartir sus confirmaciones con el repositorio remoto)



Resumen

1. Crear repositorio (ej: Github)
2. **git clone** https://github.com/my_repo
3. Crear un archivo dentro del repositorio.
4. **git add** tu_archivo o **git add .**
5. **git commit -m** “Mensaje”
6. **git push**