

# Configuración de Entorno de Desarrollo



Visual Studio Code



git

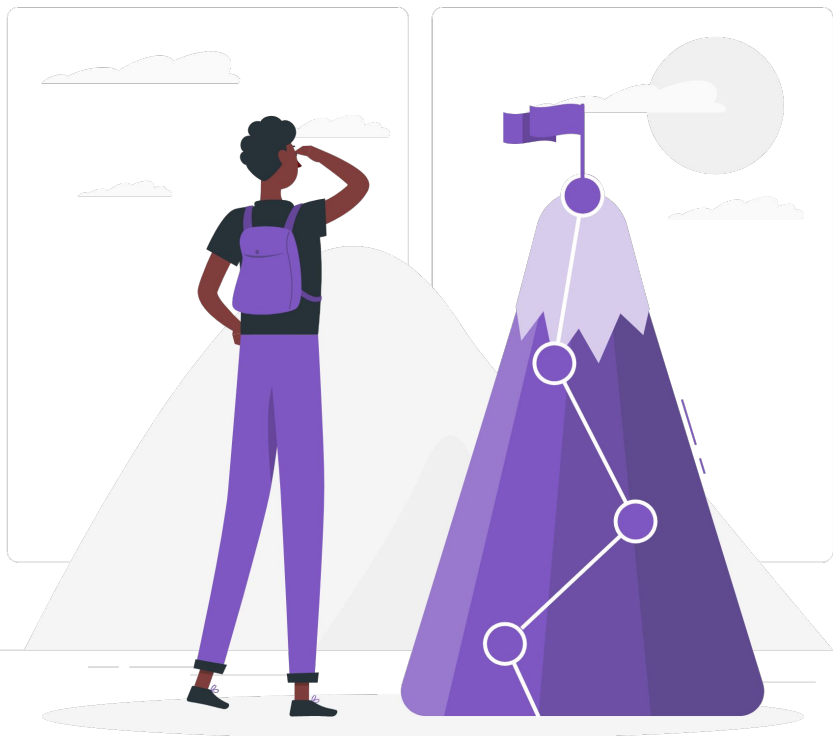
**DEV.F**  
DESARROLLAMOS(PERSONAS);



**GitHub**

Elaborado por: César Guerra

dev



# Objetivos de la Sesión

- Tener listo el entorno de desarrollo para comenzar a trabajar en el curso.
- Conocer la interfaz de Visual Studio Code.
- Añadir más funcionalidades al editor a través de Extensiones.
- Entender la importancia y el uso de la línea de comandos.
- Entender la importancia del control de versiones en software.
- Aprender a usar git y github.



Visual Studio Code

**Editor de Código:**  
**Visual Studio Code**

**DEV.F.**  
DESARROLLAMOS(PERSONAS);

dev

# Editor Recomendado: Visual Studio Code



## VS Code

[Docs](#) [Updates](#) [Blog](#) [Community](#) [Extensions](#) [FAQ](#)

Search Docs

Download

Version 1.20 is now available! Read about the new features and fixes from January.

### Code editing. Redefined.

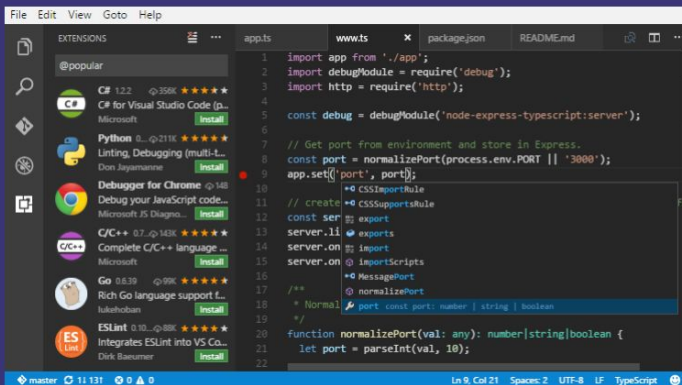
Free. Open source. Runs everywhere.

Download for Windows

Stable Build

Other platforms and Insiders Edition

By using VS Code, you agree to its  
license and privacy statement.



IntelliSense



Debugging



Built-in Git

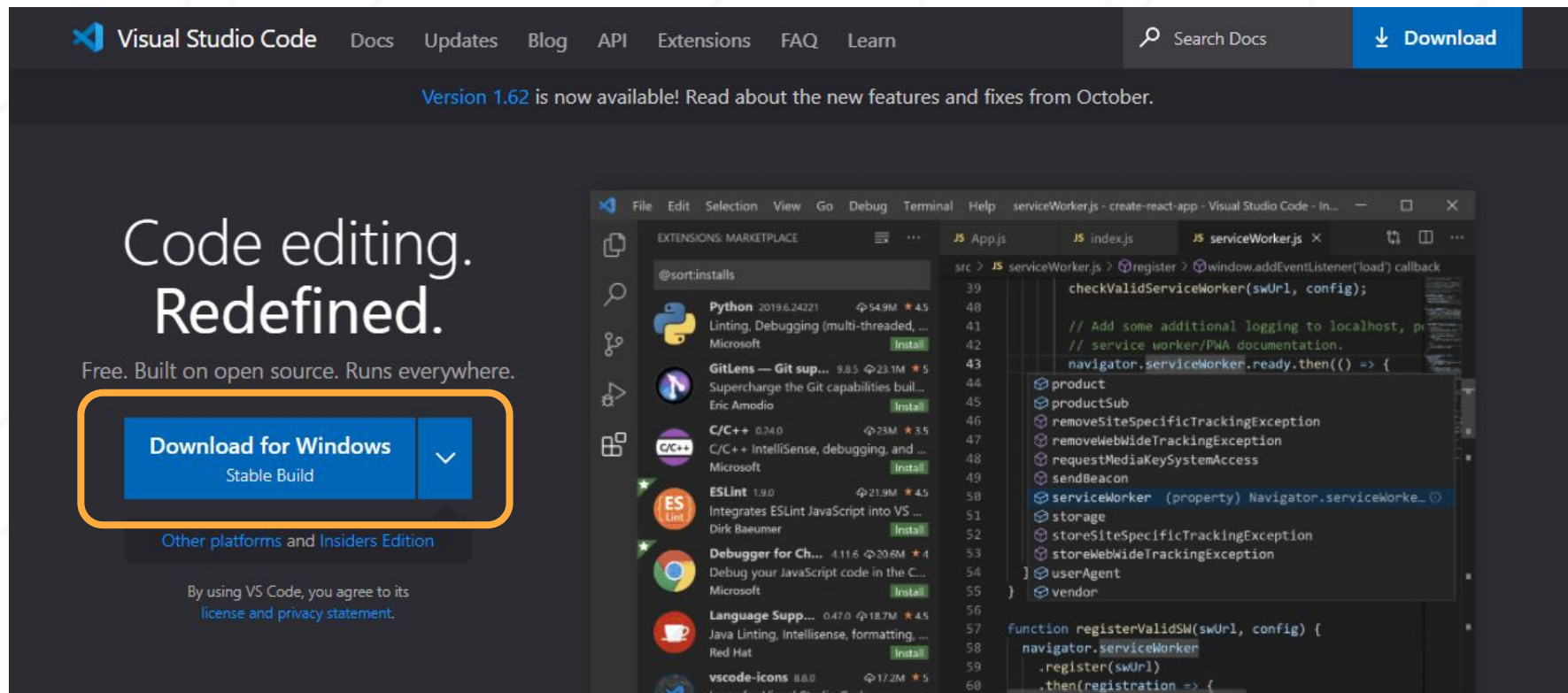


Extensions

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

DEV.F

# Descarga e Instalación de Visual Studio Code



The screenshot shows the Visual Studio Code website. At the top, there's a navigation bar with links: Visual Studio Code, Docs, Updates, Blog, API, Extensions, FAQ, Learn, a search bar labeled 'Search Docs', and a blue 'Download' button. Below the navigation bar, a message states 'Version 1.62 is now available! Read about the new features and fixes from October.' The main content area features the text 'Code editing. Redefined.' and 'Free. Built on open source. Runs everywhere.' Below this, a blue button labeled 'Download for Windows' with 'Stable Build' underneath is highlighted with an orange border. To the right of the button is a dropdown arrow. Below the button, it says 'Other platforms and Insiders Edition'. At the bottom, a small text reads 'By using VS Code, you agree to its license and privacy statement.' An inset image shows the Visual Studio Code interface with the 'EXTENSIONS: MARKETPLACE' sidebar open, displaying a list of extensions like Python, GitLens, C/C++, ESLint, and others. The main editor area shows a JavaScript file named 'serviceWorker.js' with code for a service worker.

Visual Studio Code Docs Updates Blog API Extensions FAQ Learn

Search Docs Download

Version 1.62 is now available! Read about the new features and fixes from October.

## Code editing. Redefined.

Free. Built on open source. Runs everywhere.

**Download for Windows**  
Stable Build

Other platforms and Insiders Edition

By using VS Code, you agree to its license and privacy statement.

EXTENSIONS: MARKETPLACE

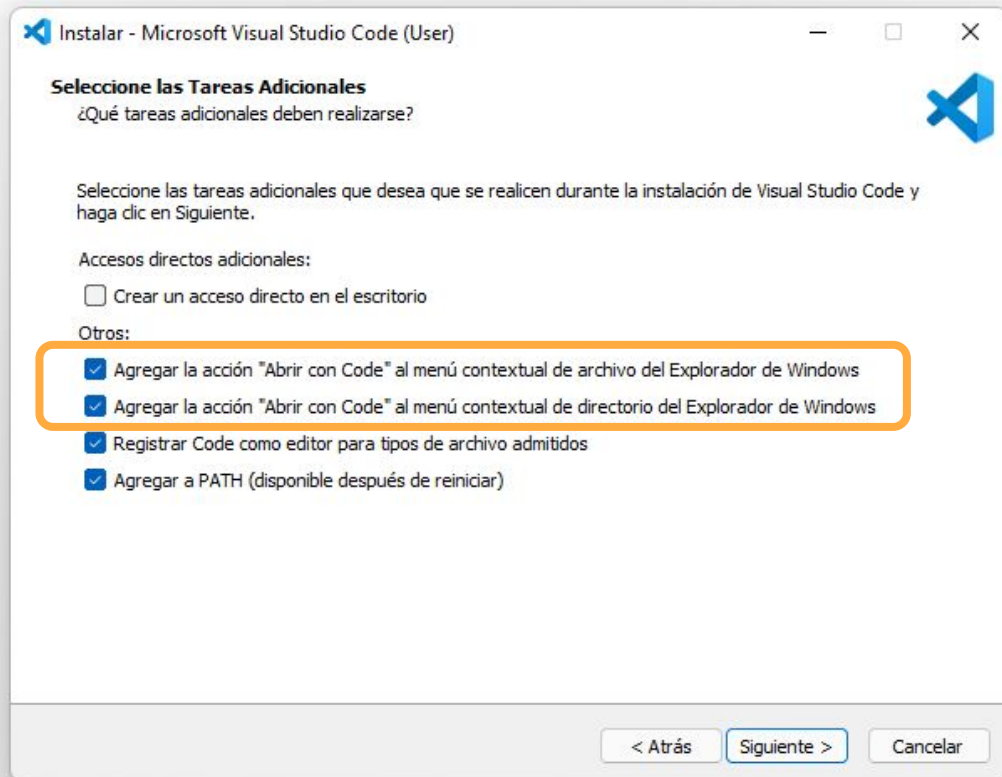
- @sortinstalls
- Python** 2019.6.24221 54.9M 4.5  
Linting, Debugging (multi-threaded, ...  
Microsoft **Install**
- GitLens — Git sup...** 9.8.5 23.1M 5  
Supercharge the Git capabilities built...  
Eric Amodio **Install**
- C/C++** 0.24.0 23M 3.5  
C/C++ IntelliSense, debugging, and ...  
Microsoft **Install**
- ESLint** 1.9.0 21.9M 4.5  
Integrates ESLint JavaScript into VS ...  
Dirk Baumer **Install**
- Debugger for Ch...** 4.11.6 20.6M 4  
Debug your JavaScript code in the C...  
Microsoft **Install**
- Language Supp...** 0.47.0 18.7M 4.5  
Java Linting, IntelliSense, formatting, ...  
Red Hat **Install**
- vscode-icons** 8.8.0 17.2M 5  
Icons for Visual Studio Code

```
src > JS serviceWorker.js > register > window.addEventListener('load') callback
39
40
41 checkValidServiceWorker(swUrl, config);
42
43 // Add some additional logging to localhost, p
44 // service worker/PWA documentation.
45 navigator.serviceWorker.ready.then(() => {
46
47   product
48   productSub
49   removeSiteSpecificTrackingException
50   removeWebWideTrackingException
51   requestMediaKeySystemAccess
52   sendBeacon
53   serviceWorker (property) Navigator.serviceWorke...
54   storage
55   storeSiteSpecificTrackingException
56   storeWebWideTrackingException
57   userAgent
58   vendor
59
60 function registerValidSW(swUrl, config) {
61   navigator.serviceWorker
62     .register(swUrl)
63     .then(registration => {
```

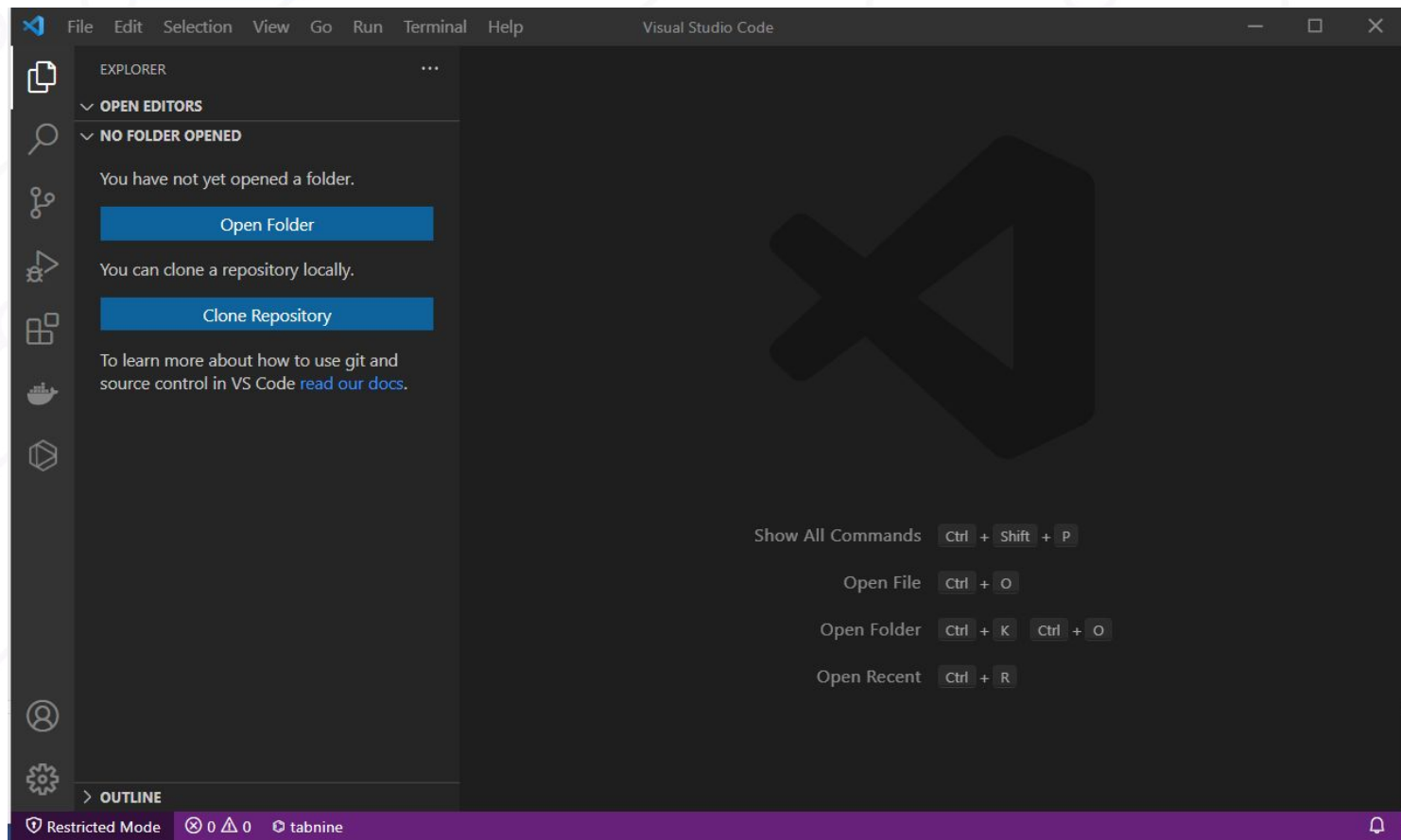
<https://code.visualstudio.com/>

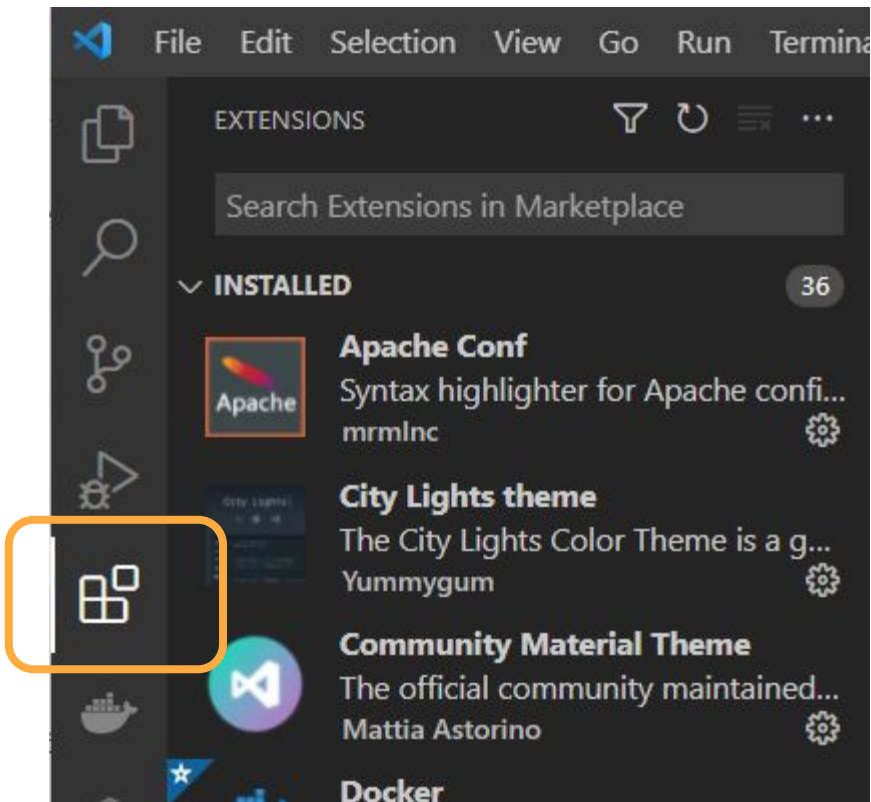
DEV.F

# Agregar “Abrir con Code” al menú contextual



# Explorando la Interfaz de VS Code





# Extensiones en VS Studio Code

Las extensiones de VS Studio Code nos permiten mejorar los super poderes del editor al proporcionarnos nuevas características.

VS Studio Code integra una función que nos permite buscar extensiones publicadas por terceros e instalarlas en nuestro editor de código.



The logo consists of the text 'DEV.F.' in a bold, white, sans-serif font. The 'F' is stylized with three small squares at its base. The logo is centered within a dark blue diamond shape.

**DEV.F.**

## **EXTENSIONES RECOMENDADAS**

*Algunas son opcionales, son completamente a gusto de cada persona, pero les recomendamos les den una oportunidad.*



## Bracket Pair Colorizer v1.0.61

CoenraadS | 6,816,869 | ★★★★★ (189)

A customizable extension for colorizing matching brackets

Disable

Uninstall



This extension is enabled globally.

### Details Feature Contributions Changelog Runtime Status

#### Enabling Bracket Color

It is recommended to disable the inbuilt `editor.matchBrackets` setting if using this feature

```
1 // Easily identify which scope you are in
2 {
3   |
4   |
5   |
6 }
```

Tip: Add the value `"backgroundColor : {color}"` to increase visibility

```
1 // Active Scope background color can increase visibility
2 {
3   |
4   |
5   |
6 }
```

"`bracketPairColorizer.showBracketsInGutter`"

Show active scope brackets in the gutter

```
{ 39 | | .map(response => {
    40 | |   return response;
  } 41 | | })
```

## Extensión Recomendada

# Bracket Pair Colorizer

Nos coloca los diferentes paréntesis, corchetes y llaves coloreados de forma diferente por cada par.

Mucho del código que escribimos se agrupa de esa forma y nos brinda mayor legibilidad para leer nuestro código.

Extension: Live Server

# Live Server

Ritwick Dey | 16,572,665 | ★★★★★ (341)

Launch a development local Server with live reload feature for static & dynamic pages

Disable Uninstall ⚙️

This extension is enabled globally.

Details Feature Contributions Changelog Runtime Status

Live Server for server side pages like PHP. [Check Here](#)

[For 'command not found error' #78]

vscode marketplace v5.6.1 downloads 20M rating 4.4/5 (341)

travis branch passing Appveyor branch license MIT

Launch a local development server with live reload feature for static & dynamic pages.

```
16 <ul>
17 <li class="current"><a href="index.html">
18 <li><a href="about.html">About</li>
19 <li><a href="services.html">Services</li>
20 </ul>
21 </div>
22 </header>
23 </div>
24 <section id="showcase">
25 <div class="container">
26 <h1>Web Design</h1>
27 <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
28 </p>
29 </div>
30 </section>
31 <section id="newsletter">
32 <div class="container">
33 <h2>Subscribe To Our Newsletter</h2>
34 <form>
35 <input type="email" placeholder="Enter Email..">
36 <button type="submit" class="button_1">Subscribe</button>
37 </form>
38 </div>
```

You're browsing as a Guest

Pages you view in this window won't appear in the browser history and they won't have other traces, like cookies, on the computer after you close all open Guest windows. Any files you download will be generated, however.

LOAD MORE

Released 27/6/2017  
on 16:14:04  
Last 17/4/2019  
updated 05:42:11  
Identifier ritwickdey.liveserver

## Extensión Indispensable Live Server

Nos permite ejecutar un servidor web que cargue el archivo que le indiquemos (normalmente el que estemos trabajando).

Si bien el navegador puede abrir directamente los archivos HTML, no es lo mismo hacerlo de esta forma que ejecutarlos desde un servidor web.



## Material Theme v33.1.2

Mattia Astorino

The most epic theme now for Visual Studio Code

Set Color Theme

Disable

Uninstall



This extension is enabled globally.

Details

Feature Contributions

Changelog

Extension Pack

Runtime Status



## Material Theme

The original one.

RATING



DOWNLOADS

562.17K

SUPPORTED BY

VSCODE POWER USER →

SUPPORTED BY

NODE CLI.COM →




## Extensión Recomendada Material Theme

Es meramente visual.

Cambia la apariencia y colores del editor. Tiene varias configuraciones.

La elección del tema suele ser personal y a gusto de cada uno.

Extension: Material Icon Theme ✕



## Material Icon Theme v4.10.0

Philipp Kief | 9,980,961 | ★★★★★ (225)

Material Design Icons for Visual Studio Code

[Set File Icon Theme](#) [Disable](#) [Uninstall](#) ⚙️

This extension is enabled globally.

[Details](#) [Feature Contributions](#) [Changelog](#) [Runtime Status](#)

### File icons

Icon	Name
	3d
	Abc
	Actionscript
	Ada
	Adonis
	Advpt_include
	Advpt_pnw
	Advpt_ptm
	Advpt_tpp
	Android
	Angular
	Angular component
	Angular directive
	Angular guard
	Angular pipe
	Angular resolver

### EXPLORER

OPEN EDITORS

14.SONGLIST-CONTEXT

- node\_modules
- public
- src
  - components
  - context
  - data
  - views \Home
    - home.css
    - Home.js
    - index.js
    - App.css

## Extensión Recomendada

# Material Icon Theme

Es meramente visual.

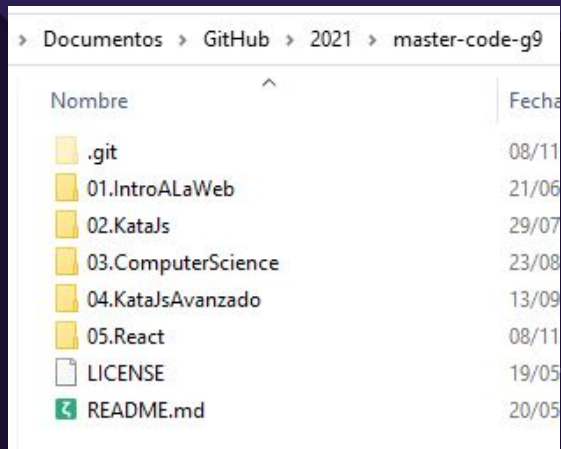
Su única función es colocar iconos a nuestros archivos y carpetas dentro del editor dependiendo de la tecnología que estemos trabajando.

# DEV.F

## TIP: Crear Carpetas para Cada Proyecto

*Resulta muy útil colocar todos los archivos relacionados a un proyecto dentro una misma carpeta.*

*Recomendamos una carpeta general del curso (master-code-g14), con subcarpetas por módulo, y entonces subcarpetas por proyecto o trabajo en el módulo.*





# Introducción a: **Terminal y Línea de Comandos**

**DEV.F**  
DESARROLLAMOS(PERSONAS);

dev



# Terminal

Un dispositivo electrónico o electromecánico que se utiliza para interactuar con una computadora.

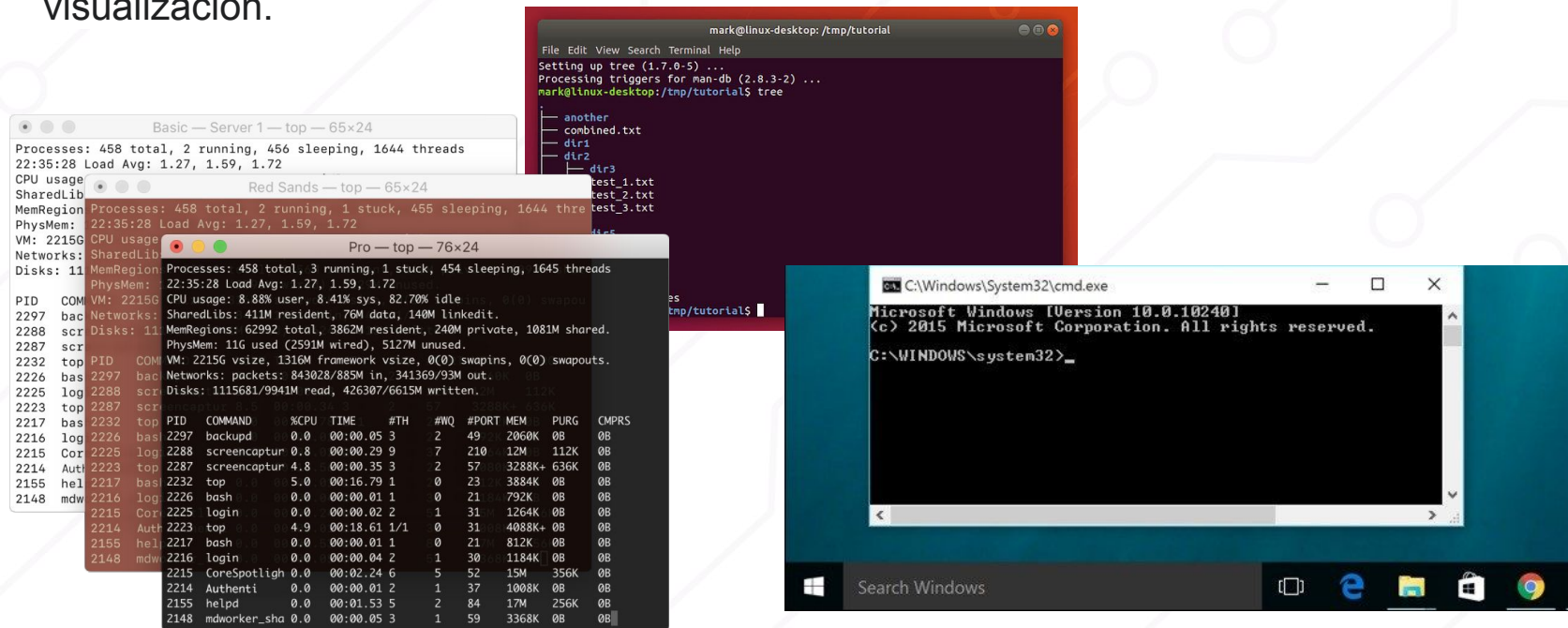
El concepto de terminal suele confundirse con el de emulador virtual, que consiste en un programa que emula las especificaciones de un terminal estándar.





# Emulador de Terminal

Un **emulador de terminal** o **emulador de consola** es un programa informático que simula el funcionamiento de una terminal de computadora en cualquier dispositivo de visualización.



# Comandos

Una orden, instrucción o comando (barbarismo del inglés *command*) es una instrucción que el usuario proporciona a un sistema informático, desde una terminal o línea de comandos o desde una llamada de programación.

Puede ser interno (contenido en el propio intérprete) o externo (contenido en un archivo ejecutable).



```
/dev/pts/12
[rickr@pizza ~]$ ls
AFNI_data6 Desktop abin afni data rbin subject_results
[rickr@pizza ~]$
[rickr@pizza ~]$ ls -la
.          .bash_logout  .gitconfig    .nvidia-settings-rc .thunderbird
..         .bash_profile .gnupg        .pki               .vim
.ICEauthority .bashrc      .gphoto       .qt                .viminfo
.Xauthority  .cache       .gststreamer-0.10 .qt-recordmydesktop AFNI_data6
.Xdefaults   .ccache      .hide.stuff    .rickrc            Desktop
.afni        .config      .history       .sbd               abin
.afni.log    .cshrc       .icons         .ssh               afni
.afni.vctime .dbus        .kde           .subversion        data
.afnirc      .esd_auth    .local         .sumarc            rbin
.anthy       .exrc        .mozilla       .themes            subject_results
.bash_history .gimp-2.8    .nv            .thumbnails
[rickr@pizza ~]$
[rickr@pizza ~]$ ls -l
total 64
drwxrwxr-x. 11 rickr users 4096 Sep 22 13:51 AFNI_data6
drwxr-xr-x.  2 rickr users 4096 Mar  9 13:05 Desktop
drwxr-xr-x.  7 rickr users 40960 Mar  9 09:16 abin
drwxr-xr-x.  9 rickr users 4096 Mar  9 13:03 afni
drwxr-xr-x. 13 rickr users 4096 Dec  8 09:33 data
drwxr-xr-x.  2 rickr users 4096 Mar  4 13:17 rbin
drwxr-xr-x.  3 rickr users 4096 Sep 22 16:24 subject_results
[rickr@pizza ~]$
```

The logo consists of the text 'DEV.F!' in a bold, white, sans-serif font. The 'F' is stylized with three small squares at its top right corner. The logo is centered within a dark blue diamond shape.

**DEV.F!**

# Comandos Básicos

*En Linux, MacOS y Windows*

dev

# Mostrar contenido de una carpeta

## Linux / Mac OS

### **ls** (list)

Imprime el contenido de un directorio.

Como casi todos los comandos, tiene variaciones como

- `ls -a`
- `ls -l`

`$ ls`

## Windows

### **dir** (directory)

Muestra todos los directorios y archivos contenidos dentro de un directorio

`$ dir`

# Saber dónde estoy...

## Linux / Mac OS

**pwd**  
(*print working directory*)

Este comando imprime la ubicación de tu directorio de trabajo actual

Es importante saber dónde te encuentras antes de ir a un directorio principal o secundario.

`$ pwd`

## Windows

**cd**  
(*current directory*)

Sin pasarle ningún parámetro adicional, cd te muestra tu ubicación actual

`$ cd`

También es posible usar  
`$ echo %cd%`

# Movernos entre carpetas (cambiar de ruta)

## Linux / Mac OS

**cd /directorio**  
(*change directory*)

Permite moverte a otro directorio

`$ cd Documents/DEVF`

## Windows

**cd /directorio**  
(*change directory*)

Si pasamos un parámetro adicional a solo escribir `cd`, podemos movernos a otro directorio

`$ cd Documents/DEVF`

Linux / Mac OS

## **mkdir** (*make directory*)

El comando *mkdir* se utiliza para crear un nuevo directorio

```
$ mkdir mi_directorio
```

Windows

## **mkdir** (*make directory*)

El comando *mkdir* se utiliza para crear un nuevo directorio

```
$ mkdir mi_directorio
```

# Copiar archivos / carpetas

## Linux / Mac OS

**cp**  
(*copy*)

Sirve para copiar archivos y directorios.

```
$ cp archivo_a archivo_a_copia
```

## Windows

**copy**

Nos permite copiar archivos y directorios.

```
$ copy archivo_a archivo_a_copia
```



# Mover archivos / carpetas

## Linux / Mac OS

### **mv** (*move*)

El comando *mv* se usa para mover o renombrar directorios y archivos.

```
$ mv nombre_viejo nombre_nuevo
```

## Windows

### **move**

El comando *mv* se usa para mover o renombrar directorios y archivos.

```
$ move nombre_viejo nombre_nuevo
```

# Borrar archivos

## Linux / Mac OS

**rm**  
(*remove*)

El comando *rmdir* se utiliza para eliminar un archivo.

`$ rm mi_directorio`

## Windows

**del**  
(*delete*)

El comando *rmdir* se utiliza para eliminar un archivo

`$ del archivo.txt`



# Borrar carpetas

## Linux / Mac OS

**rmdir**  
(remove directory)

El comando *rmdir* se utiliza para eliminar un directorio

```
$ rmdir mi_directorio
```

## Windows

**rmdir**  
(remove directory)

El comando *rmdir* se utiliza para eliminar un directorio

```
$ rmdir mi_directorio
```

# Imprimir en pantalla el contenido de un archivo

## Linux / Mac OS

**cat**  
(*concatenate*)

Se utiliza para imprimir el contenido de un archivo en la pantalla, útil cuando deseas verlo rápidamente

```
$ cat main.js
```

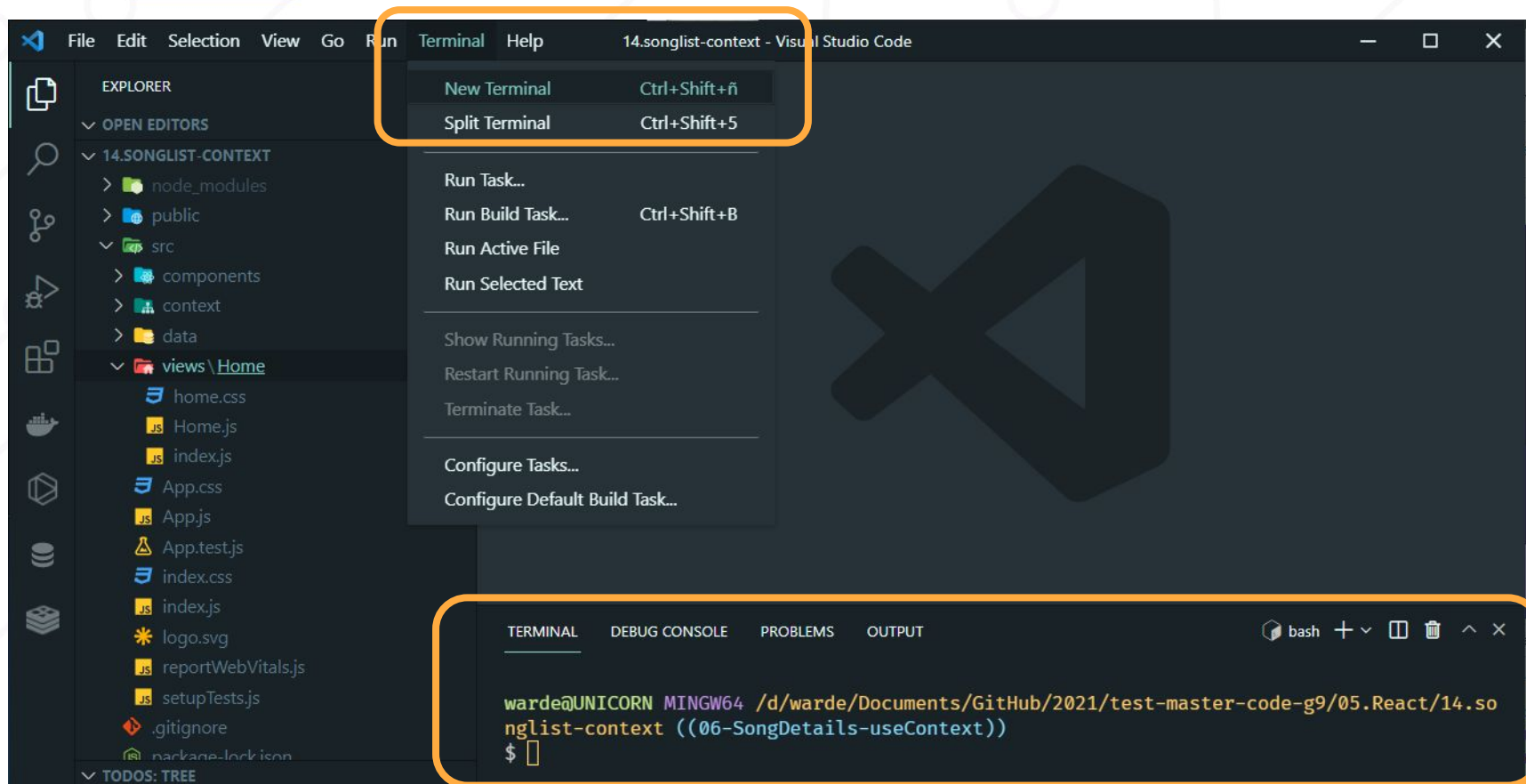
## Windows

**type**  
(*escribir / teclear*)

Se utiliza para imprimir el contenido de un archivo en la pantalla, útil cuando deseas verlo rápidamente

```
$ type main.js
```

# VS Studio Code Incluye una Terminal





# Sistema de Control de Versiones: Git

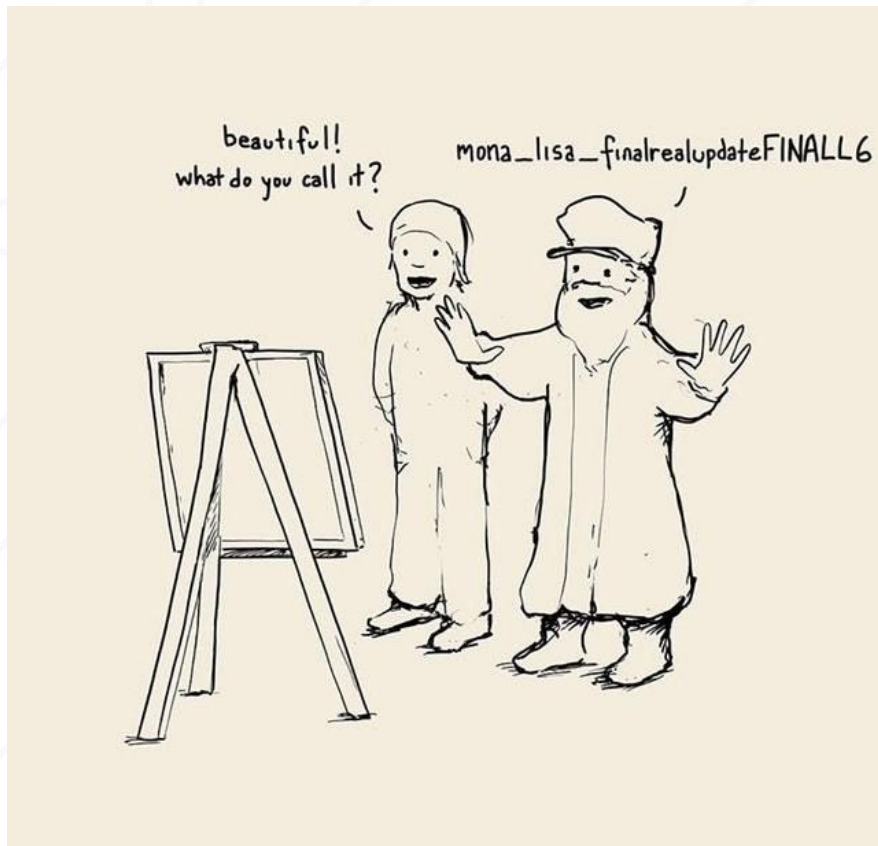
**DEV.F.**  
DESARROLLAMOS(PERSONAS);

**DEV.F**



**Git es un  
Sistema de Control  
de Versiones**

# ¿Que significa controlar una versión?



- segunda revision
- tesis-corregida
- tesis-corregida copy
- tesis-revisada
- tesis-v-finañ
- tesis1



Podemos cambiar la historia...



# Sistema de control de versiones (VCS)

Un **sistema de control de versiones (VCS - Version control system)** es aquel que nos permite llevar un historial y control de cambios a medida que las personas y los equipos colaboran en proyectos juntos.



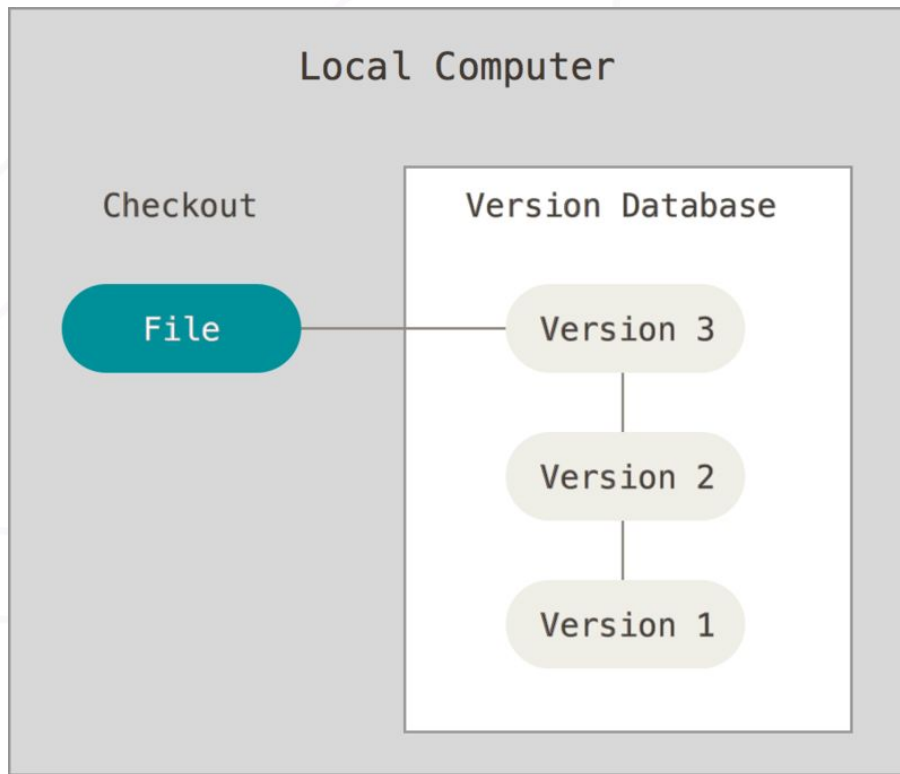
¿Que cambios se hicieron?

¿Quién hizo los cambios?

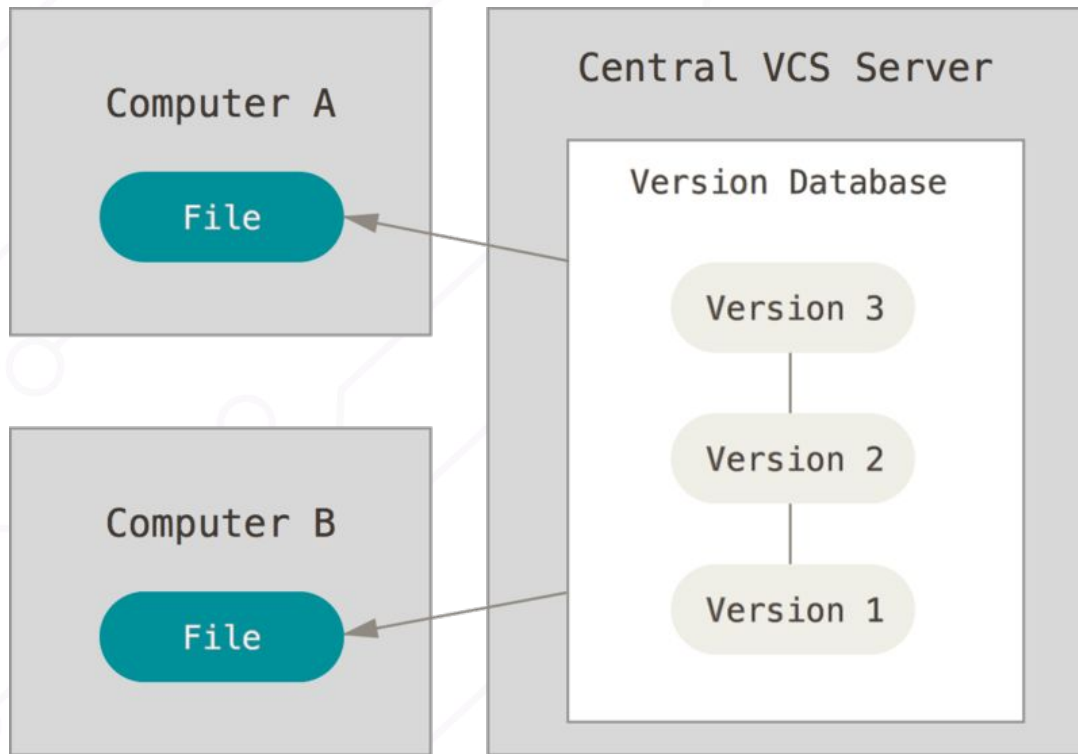
¿Cuando se hicieron los cambios?

¿Por qué fueron requeridos los cambios?

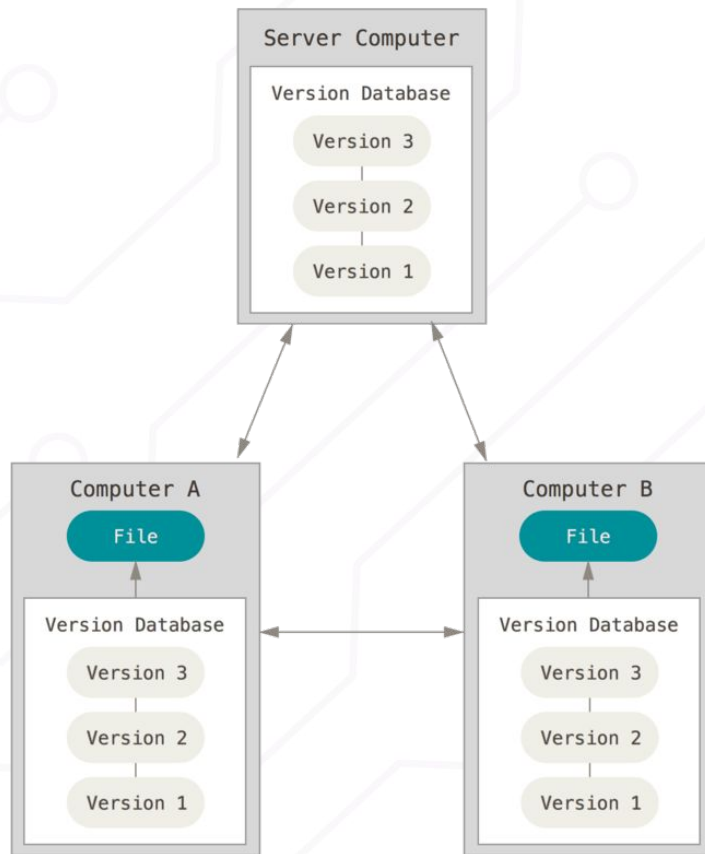
# Tipos de VCS (Local)



# Tipos de VCS (Centralizado)



# Tipos de VCS (Distribuido)





**Git** es un (VCS) de tipo **distribuido** de código abierto y actualmente el más usado por los desarrolladores gracias a sus beneficios para individuos y equipos de trabajo como:

- Acceso detallado a la historia del proyecto.
- Colaboración en cualquier momento y lugar.

Su uso principal es mediante Interfaz de línea de comandos (**CLI - Command line interface**)

# ¿Qué es un repositorio?



Un **repositorio** es un espacio de almacenamiento donde se organiza, mantiene y difunde información.

En el caso de **Git** el **repositorio** viene siendo el proyecto donde estará la colección de archivos y carpetas junto al historial de cambios.



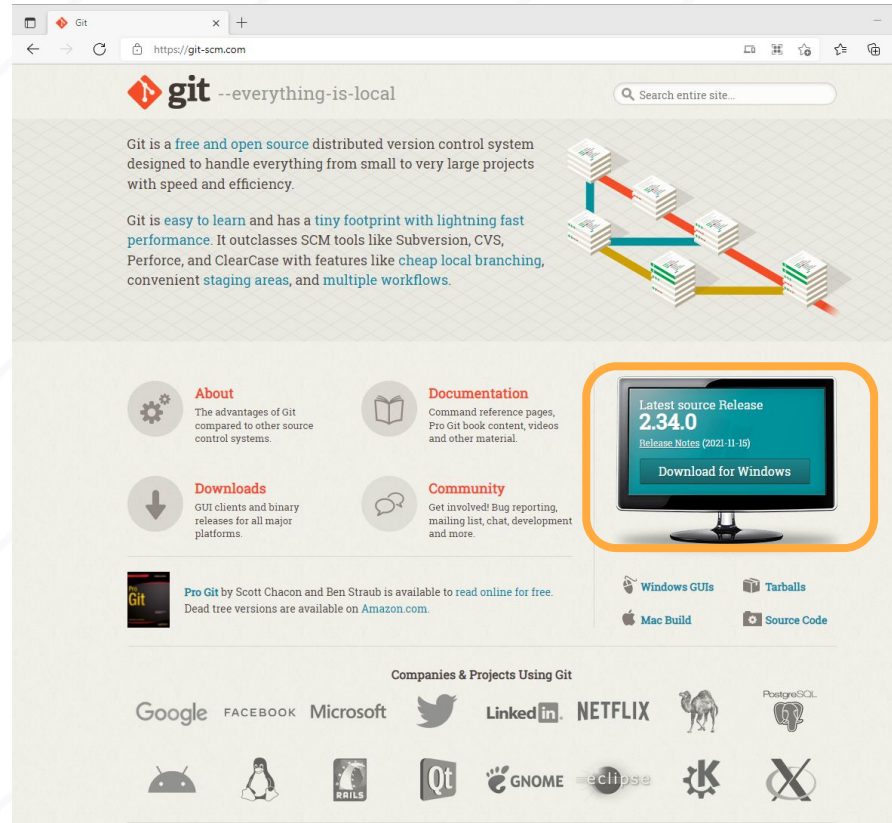


# Instalación de Git



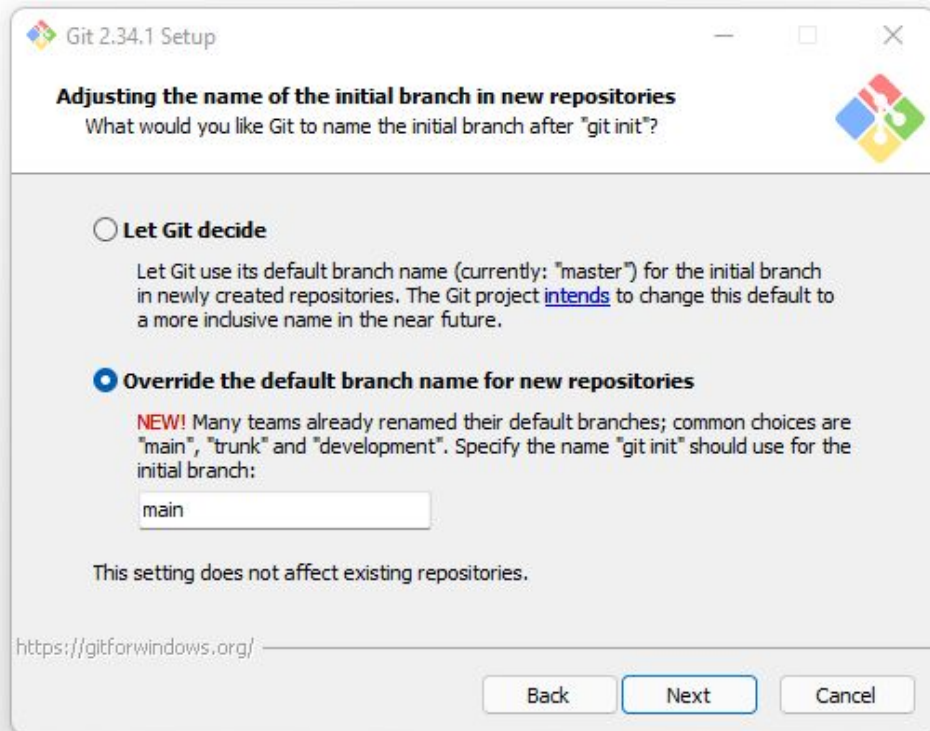
# git

<https://git-scm.com/>





# Durante la instalación: Rama main



# Configuración de Git

- Desde consola, se puede acceder a la configuración de **Git** con el comando:

```
git config
```

- Se recomienda establecer una identidad en **Git**, para ello se usan los comandos:

```
git config user.name  
git config user.email
```

- Usando el flag “**--global**” podemos establecer la configuración de forma global y realizarla una sola vez.

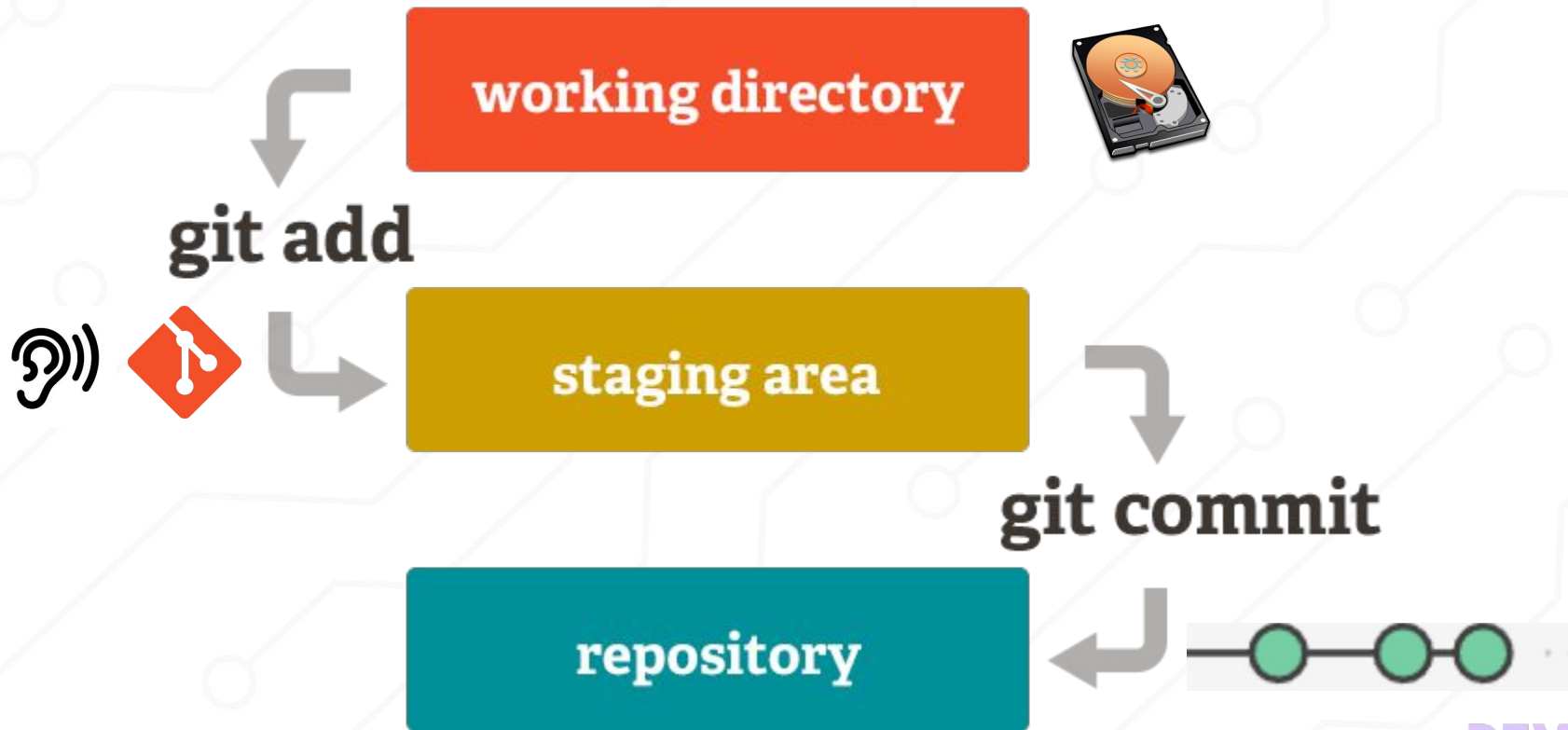
```
git config --global user.name "César Guerra"  
git config --global user.email "cesar@devf.mx"
```



Nota: Podemos verificar la configuración actual con:

```
git config --list
```

# Estados de Git





# Pasos para tomar una Foto en la Playa

**Paso 1:** Decidir ir a la playa y estar ahí.

**Paso 2:** Con una cámara, realizar un encuadre de los elementos que queremos que aparezca en la foto.

**Paso 3:** Presionar el botón de tomar fotografía.



# Pasos para tomar una Foto en la Playa

**Paso 1:** Decidir ir a la playa y estar ahí

**working directory** (`git init` / `git clone`)

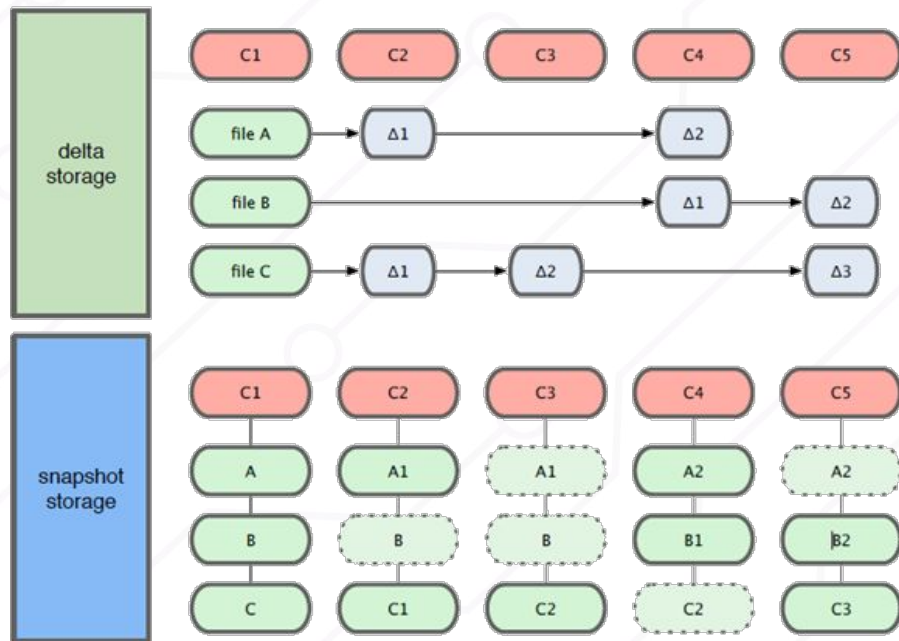
**Paso 2:** Con una cámara, realizar un encuadre de los elementos que queremos que aparezca en la foto.

**staging area** (`git add`)

**Paso 3:** Presionar el botón de tomar fotografía.

**repository** (`git commit`)

# Instantáneas (Snapshots)



Un **Snapshot** es un estado de 'algo' en el tiempo, **Git** maneja **Snapshots** no cambios.

# Comandos básicos de Git

**git init** -> Inicializamos repositorio (lo usamos una sola vez al crear un nuevo proyecto, dentro de la carpeta del proyecto)

**git status** -> Nos muestra el estado de working y staging area..

**git add .** -> Agregamos todos los archivos al staging area.

**git add archivo.txt** -> Agregamos el archivo.txt al staging area.

**git commit -m "Comentario"** -> Se crea un punto en la historia con un mensaje.

**git reset archivo.txt** -> Quita el archivo.txt del staging area.

**git reset** -> Quita todos los archivos del staging area.

The logo consists of the text 'DEV.F.' in a bold, white, sans-serif font. The 'F' is stylized with three small squares at its top right corner. The logo is centered within a dark blue diamond shape.

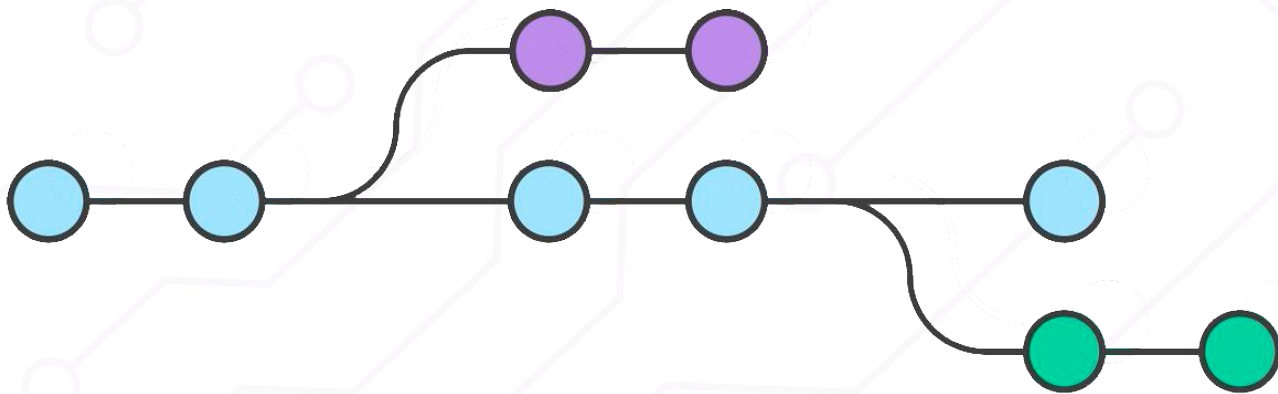
**DEV.F.**

# **Demostración Git básico**



# Ramas (Branch)

Las **ramas (Branch)** son bifurcaciones o variantes de un repositorio, estas pueden contener diferentes archivos y carpetas o tener todo igual excepto por algunas líneas de código.



# Comandos de Manejo de Ramas de Git

`git branch` -> Mostrar las ramas que tenemos.

`git branch newBranchName` -> Creamos una nueva rama.

`git checkout nombre` -> Nos cambiamos a la rama nombre.

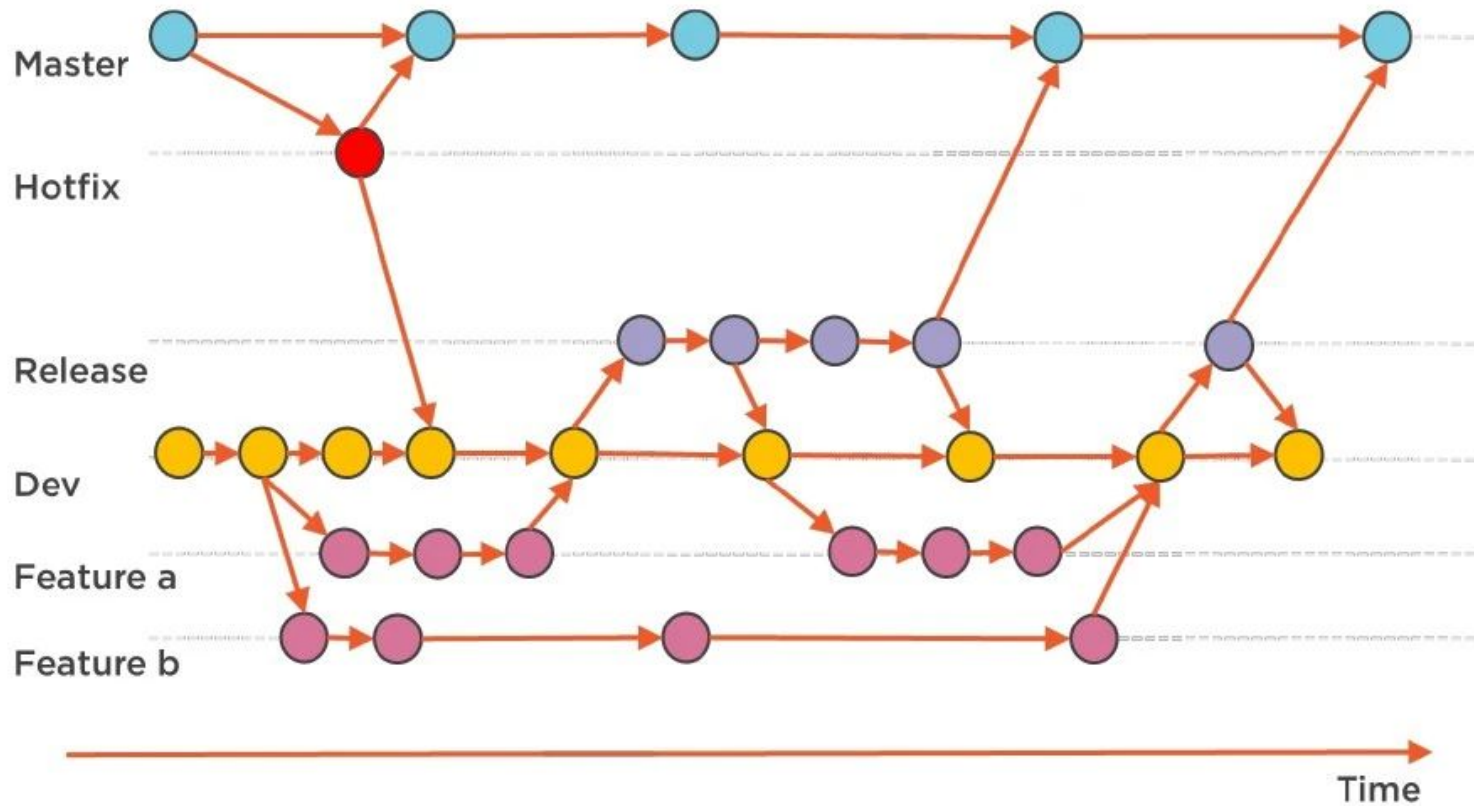
`git checkout -b nombre` -> Crear y cambiarse a una nueva rama.

`git checkout hash` -> Cambiarnos a un commit en específico.

`git checkout .` -> Regresar al commit más reciente de la rama actual.

`git merge sourceBranch` -> Unimos cambios de una rama.

# Git Flow





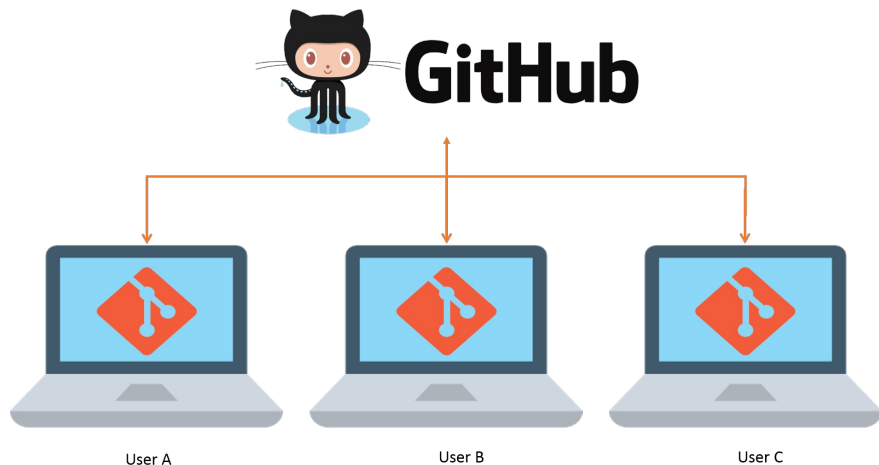
**GitHub**

**Repositorio Remoto**

**Github**

**DEV.F.**  
DESARROLLAMOS(PERSONAS);

dev



<https://github.com/>

## Github

GitHub es un servicio basado en la nube que aloja un sistema de control de versiones (VCS) llamado Git.

Éste permite a los desarrolladores colaborar y realizar cambios en proyectos compartidos, a la vez que mantienen un seguimiento detallado de su progreso.

Es decir, github hace uso de git para permitirnos tener el código almacenado en un lugar centralizado y poder colaborar.

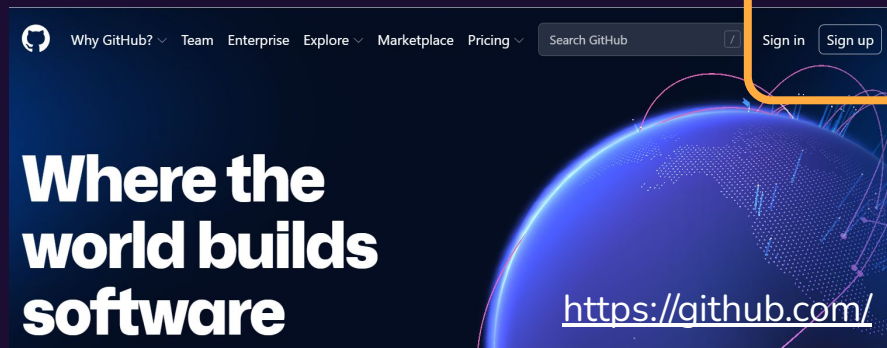


# La importancia de Aprender Git y GitHub.

DEV.F



## Creación de una cuenta de Github.



# Conectar un repositorio local a uno remoto

Mediante el comando **git remote add origin**.

## Diferencias entre el repo remoto y local

- El historial de cambios es gestionado por GIT.
- Los repos remotos son el respaldo de nuestro local.

Es necesario crear un repositorio en github para poder enlazarlo al repositorio local. Es importante saber donde será inicializado el repositorio (si local o en github)



# Comandos para Trabajar Repositorios Remotos

`git remote add origin url` -> Agregar la conexión de nuestro repo local al remoto.

`git remote set-url aliasName myNewUrl` -> Agregar la conexión de nuestro repo local al remoto.

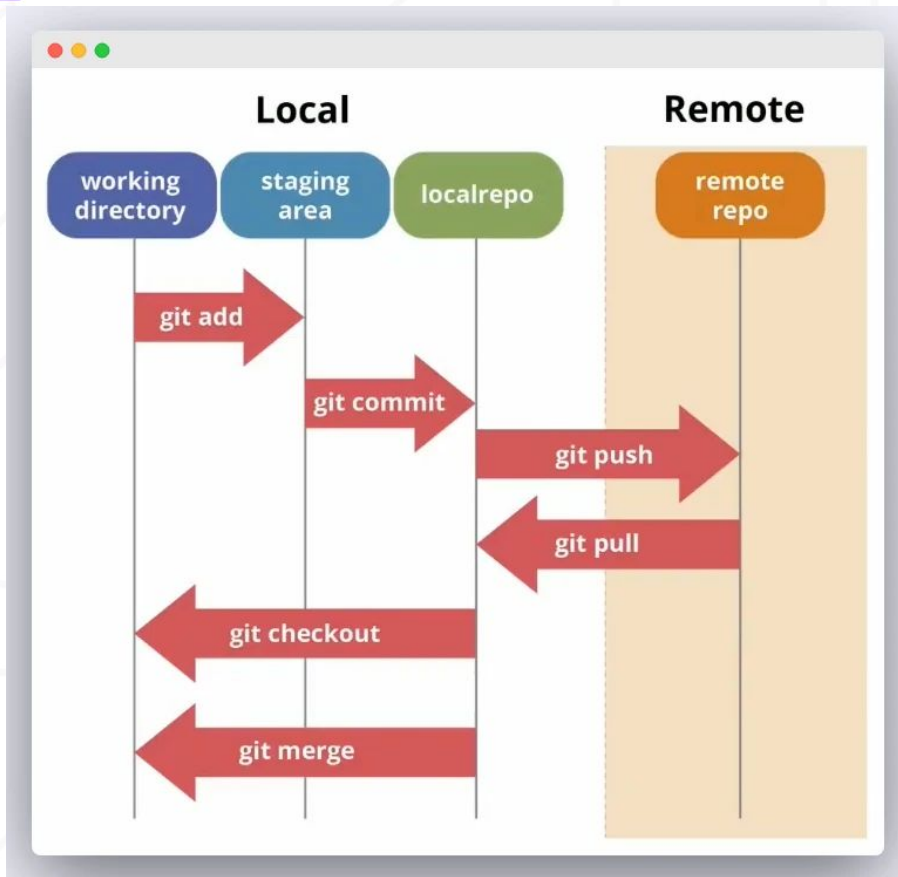
`git clone url` -> Clonar repositorio existente.

`git push alias branch` -> Enviamos cambios a repositorio remoto. Ejemplo: `git push origin main`

`git push --all origin` -> Subir todas las ramas desde local a remoto.

`git pull alias branch` -> Obtenemos cambios más recientes de la rama. Ejemplo: `git pull origin main`




# Flujo con repositorio remoto



## En caso de Incendio...

In case of fire



-  1. git commit
-  2. git push
-  3. exit building

The logo for DEV.F. is displayed in white, bold, sans-serif capital letters. The 'F' is stylized with three small squares at its top right corner. It is centered within a dark blue diamond shape.

**DEV.F.**

**Demostración  
Github**