

Qué es un sistema de control de versiones?

Clasificación de los sistemas de control de versiones

GIT

Flujo de trabajo

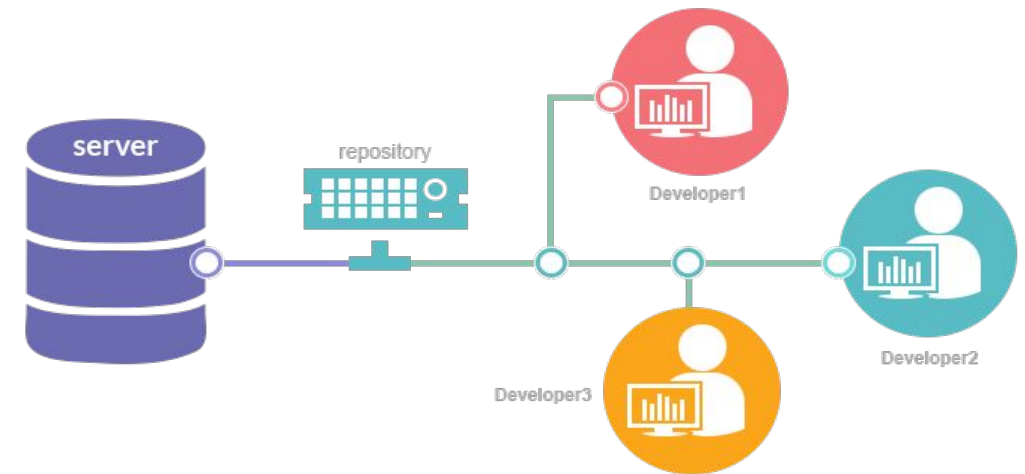
Configuración

Comandos

GitHUD

Qué es un sistema de control de versiones?

- Sistema que registra los cambios realizados sobre un archivo o conjunto de archivos a lo largo del tiempo, de modo que puedas recuperar versiones específicas más adelante.
- Permite revertir archivos a un estado anterior, revertir el proyecto entero a un estado anterior, comparar cambios a lo largo del tiempo, ver quién modificó por última vez algo que puede estar causando un problema, quién introdujo un error y cuándo, entre otras cosas.
- Aplica para cualquier tipo de archivos, no necesariamente código fuente.



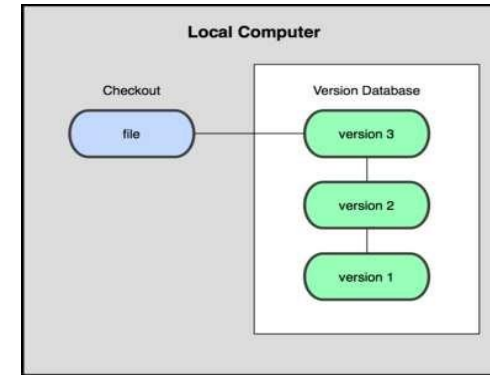
- Trabajo en equipo
- Historial de cambios
- Comparación de código
- Fusión de código



Sistema de control de versiones locales

Un método usado por muchos es copiar los archivos a otro directorio. Este enfoque es muy común porque es muy simple, pero también propenso a errores.

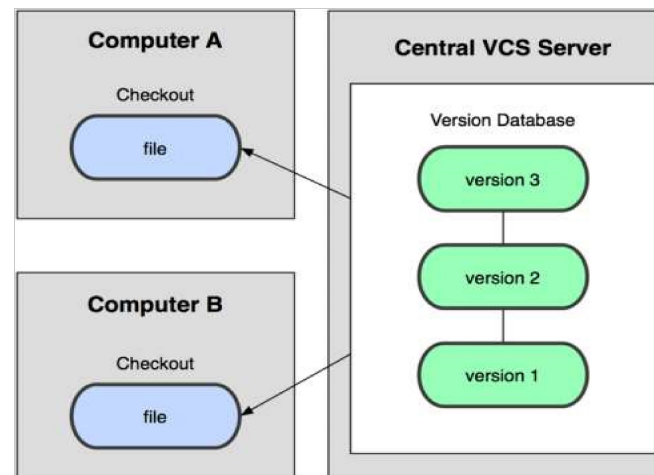
Ejemplo :RCS



Sistema de control de versiones centralizados (Centralized Version Control Systems o CVCSs)

Un único servidor que contiene todos los archivos versionados, y varios clientes que descargan los archivos desde ese lugar central

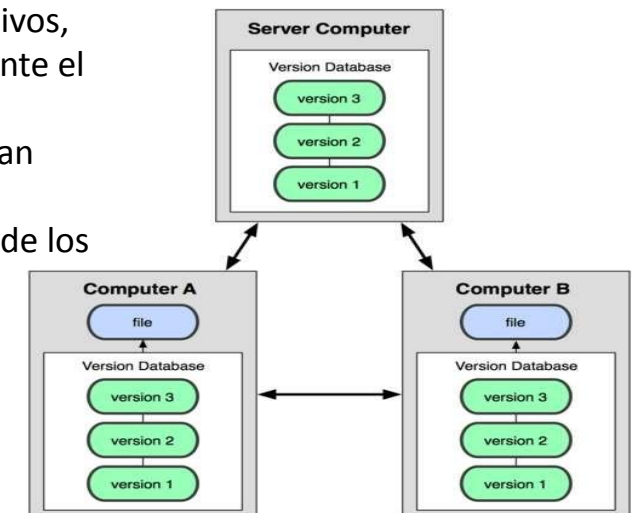
Ejemplo : TortoiseSVN



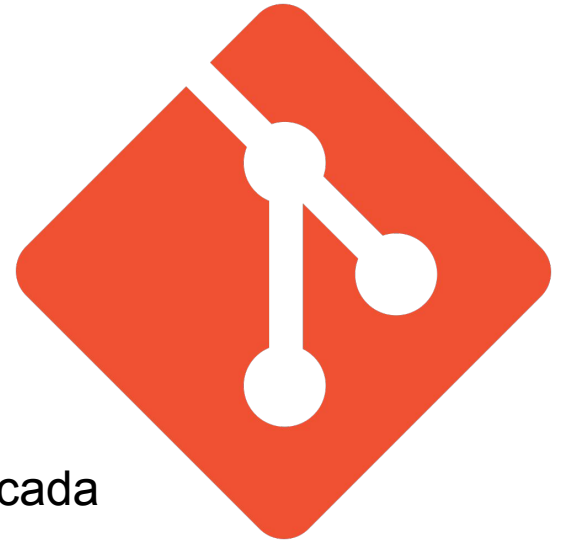
Sistema de control de versiones distribuidos (Distributed Version Control Systems o DVCSs)

Los clientes no sólo descargan la última instantánea de los archivos, también replican completamente el repositorio. Así, si un servidor muere, y estos sistemas estaban colaborando a través de él, cualquiera de los repositorios de los clientes puede copiarse en el servidor para restaurarlo.

Ejemplo Git



Git

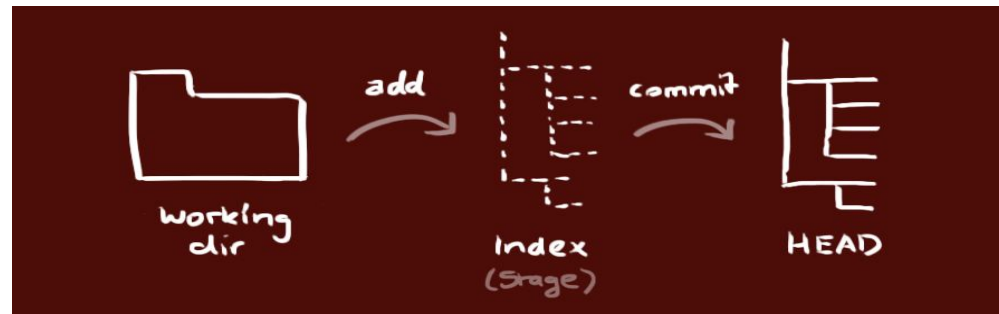


- Nació en 2005
- Otros sistemas tienden a almacenar los datos como cambios de cada archivo respecto a una versión base.
- Git modela sus datos más como un conjunto de instantáneas de un mini sistema de archivos. Cada vez que confirmas un cambio, o guardas el estado de tu proyecto en Git, él básicamente hace una foto del aspecto de todos tus archivos en ese momento, y guarda una referencia a esa instantánea.

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

Flujo de trabajo

Tu repositorio local está compuesto por tres "árboles" administrados por git. El primero es tu **Directorio de trabajo** que contiene los archivos, el segundo es el **Index** que actúa como una zona intermedia, y el último es el **HEAD** que apunta al último commit realizado



Configuración de GIT



Configurar perfil

```
git config --global user.name "Ignacio"
```

```
git config --global user.email "ignacio.camillucci@bue.edu.ar"
```

Verificar

```
git config --global -e (la -e es para abrir un editor)
```


Comandos GIT

Iniciar repositorio

git init

Verificar estado del repositorio

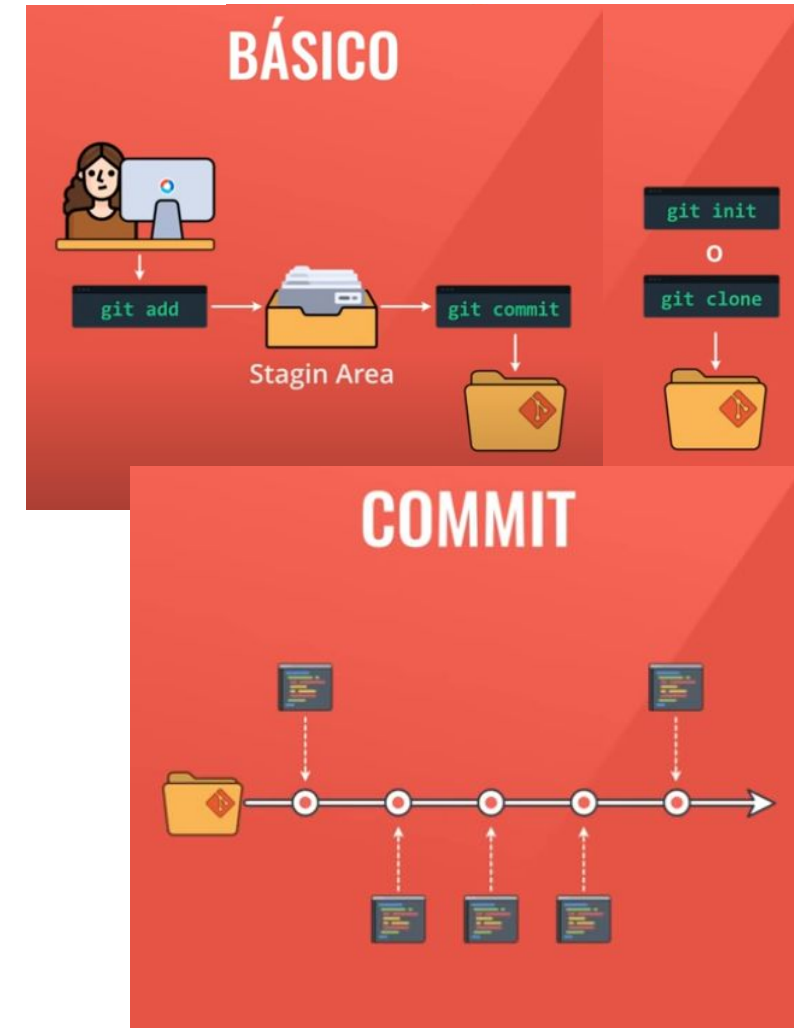
git status

Agregar archivos stage

git add . *(con el punto estamos agregando todos los archivos)*

Commit

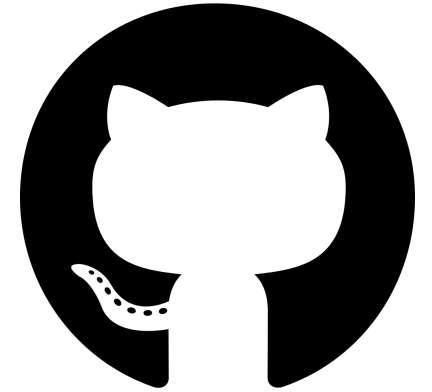
git commit -m "Mensaje primer commit"




GITHUB

Github es un servicio que nos permite tener nuestro repositorio de manera remota es decir que podemos subir nuestro repositorio local a github para poder utilizarlo en cualquier momento desde cualquier dispositivo que se encuentre conectado a internet.

<https://github.com/>





Sign in to GitHub

Username or email address

Password

[Forgot password?](#)

Sign in

New to GitHub? [Create an account.](#)

github.com

Search or jump to...

Pull requests Issues Marketplace Explore

Incognito

Repositories

New

Find a repository...

igncam/java1

JuanFalabrino/Alumnos2020

JuanFalabrino/prueba

Working with a team?

GitHub is built for collaboration. Set up an organization to improve the way your team works together, and get access to more features.

Create an organization

Discover interesting projects and people to populate your personal news feed.

Your news feed helps you keep up with recent activity on repositories you [watch](#) and people you [follow](#).

Explore GitHub

ProTip! The feed shows you events from people you follow and repositories you watch.

Subscribe to your news feed

Explore repositories

openai/jukebox

Code for "Jukebox: A Generative Model for Music"

Python 1.4k

foss-responders/fossresponders.com

The website for FOSS Responders

JavaScript 15

linq2db/linq2db

Linq to database provider.

C# 1.7k

Explore more →

© 2020 GitHub, Inc.

Blog

About

Shop

Contact GitHub

API

Training

Status

Security

Terms

Privacy

Help

Create a new repository

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

[Import a repository.](#)

Owner



JuanFalabrino

Repository name *

/

Great repository names are short and memorable. Need inspiration? How about **turbo-system**?

Description (optional)

☒ **Public**

Anyone can see this repository. You choose who can commit.

☐ **Private**

You choose who can see and commit to this repository.

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

☐ **Initialize this repository with a README**

This will let you immediately clone the repository to your computer.

Add .gitignore: None

Add a license: None



Create repository

JuanFalabrino / prueba2 Private

Unwatch 1

Star 0

Fork 0

Code

Issues 0

Pull requests 0

Actions

Projects 0

Security 0

Insights

Settings

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

[Set up in Desktop](#) or [HTTPS](#) [SSH](#) <https://github.com/JuanFalabrino/prueba2.git>

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 "# prueba2" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/JuanFalabrino/prueba2.git
git push -u origin master
```

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

```
git remote add origin https://github.com/JuanFalabrino/prueba2.git
git push -u origin master
```

...or import code from another repository

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

[Import code](#)

The screenshot shows the GitHub repository settings page for a repository named 'prueba2' owned by 'JuanFalabrino'. The repository is marked as 'Private'. The top navigation bar includes links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. The repository header shows 'Unwatch' (1), 'Star' (0), and 'Fork' (0) buttons. The left sidebar contains a list of settings: 'Options', 'Manage access' (highlighted), 'Branches', 'Webhooks', 'Notifications', 'Integrations', 'Deploy keys', 'Secrets', and 'Actions'. The main content area is divided into two sections: 'Who has access' and 'Manage access'. The 'Who has access' section shows two access levels: 'PRIVATE REPOSITORY' (locked) and 'DIRECT ACCESS' (unlocked). The 'PRIVATE REPOSITORY' section states 'Only those with access to this repository can view it.' and has a 'Manage' link. The 'DIRECT ACCESS' section states '0 collaborators have access to this repository. Only you can contribute to this repository.' and has a 'Manage' link. The 'Manage access' section shows a message 'You haven't invited any collaborators yet' with an 'Invite a collaborator' button. The footer contains copyright information for GitHub, Inc. and links to 'Terms', 'Privacy', 'Security', 'Status', 'Help', 'Contact GitHub', 'Pricing', 'API', 'Training', 'Blog', and 'About'.

Search or jump to... Pull requests Issues Marketplace Explore

JuanFalabrino / prueba2 Private

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Actions Projects 0 Security 0 Insights Settings

Options

Manage access

Branches

Webhooks

Notifications

Integrations

Deploy keys

Secrets

Actions

Who has access

PRIVATE REPOSITORY

Only those with access to this repository can view it.

Manage

DIRECT ACCESS

0 collaborators have access to this repository. Only you can contribute to this repository.

Manage

Guardar Ctrl+G

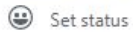
Manage access

You haven't invited any collaborators yet

Invite a collaborator

© 2020 GitHub, Inc. Terms Privacy Security Status Help

Contact GitHub Pricing API Training Blog About

[Edit profile](#)

ProTip! Updating your profile with your name, location, and a profile picture helps other GitHub users get to know you.

 [Edit profile](#)

X

Repositories 2

Projects 0

Packages 0

Stars 0

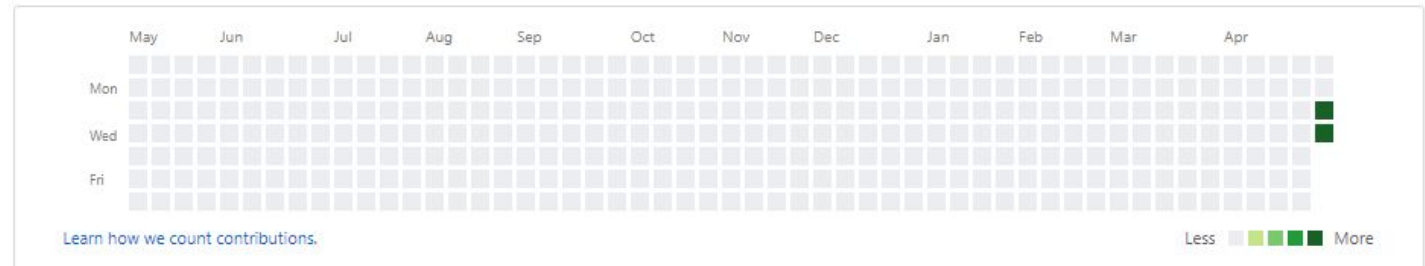
Followers 0

Following 0

Customize your pins

Alumnos2020

Contribution settings ▾



2020

Crear el README

```
echo "# prueba" >> README.md  
git commit -m "first commit"
```

Push de Nuestro proyecto

```
git remote add origin https://github.com/JuanFalabrino/prueba.git  
git push -u origin master
```

