

Git!



Git! ¿que es?

- Sistema de control de versiones
- ¿Qué es el control de versiones, y por qué debería importarte? El control de versiones es un 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.
- Cualquier tipo de archivo que encuentres en un ordenador puede ponerse bajo control de versiones.
- <https://git-scm.com/book/es/v1/Empezando-Acerca-del-control-de-versiones>



Git – objetivo de esta presentación

- Mostrar su potencial para diversos usos fuera del mundo del desarrollo de software.
- Por esto no voy a mostrar algunas de sus características destacadas para los desarrolladores.
- Si eres desarrollador y no conoces git, esta será una breve introducción útil.

Origen de Git y Objetivo

- Creado por Linus Trovalds y otros involucrados en la comunidad de desarrollo de Linux.
- <https://git-scm.com/book/es/v1/Empezando-Una-breve-historia-de-Git>

Algunos de los objetivos:

- Velocidad
- Diseño simple
- Fuerte soporte sobre desarrollo no-lineal (miles de ramas en paralelo)
- Distribuido.
- Capaz de manejar grandes proyectos como el kernel de Linux con eficiencia, velocidad y tamaño de datos.

Git! Usos!

- Colaborar en proyectos de desarrollo.
- Colaborar con cualquier equipo.
- Git es muy eficiente para sincronizar binarios tambien.
- Git se puede usar para respaldos locales o remotos.
- Git se puede usar para control de version local.

Git uso local & remoto

- Local sin sincronizar (solo controlar versiones)
 - `git init`
- Local a otro directorio en el sistema
 - `git remote add local1 /algun/otro/repo.git`
 - (`repo.git` debe haber sido iniciado con `git init - bare`)
- Remoto a un servidor propio, solo usando ssh :)
 - `Git remote add origen`
`usuario@host:/alugn/otro/repo.git`

Git Hosting conocido

- Github: 38+ millon projects.



- Gitlab



- Bitbucket (de atlassian)



Tutoriales para iniciar

- <http://rogerdudler.github.io/git-guide/index.es.html>

crea un repositorio nuevo

Crea un directorio nuevo, ábrelo y ejecuta

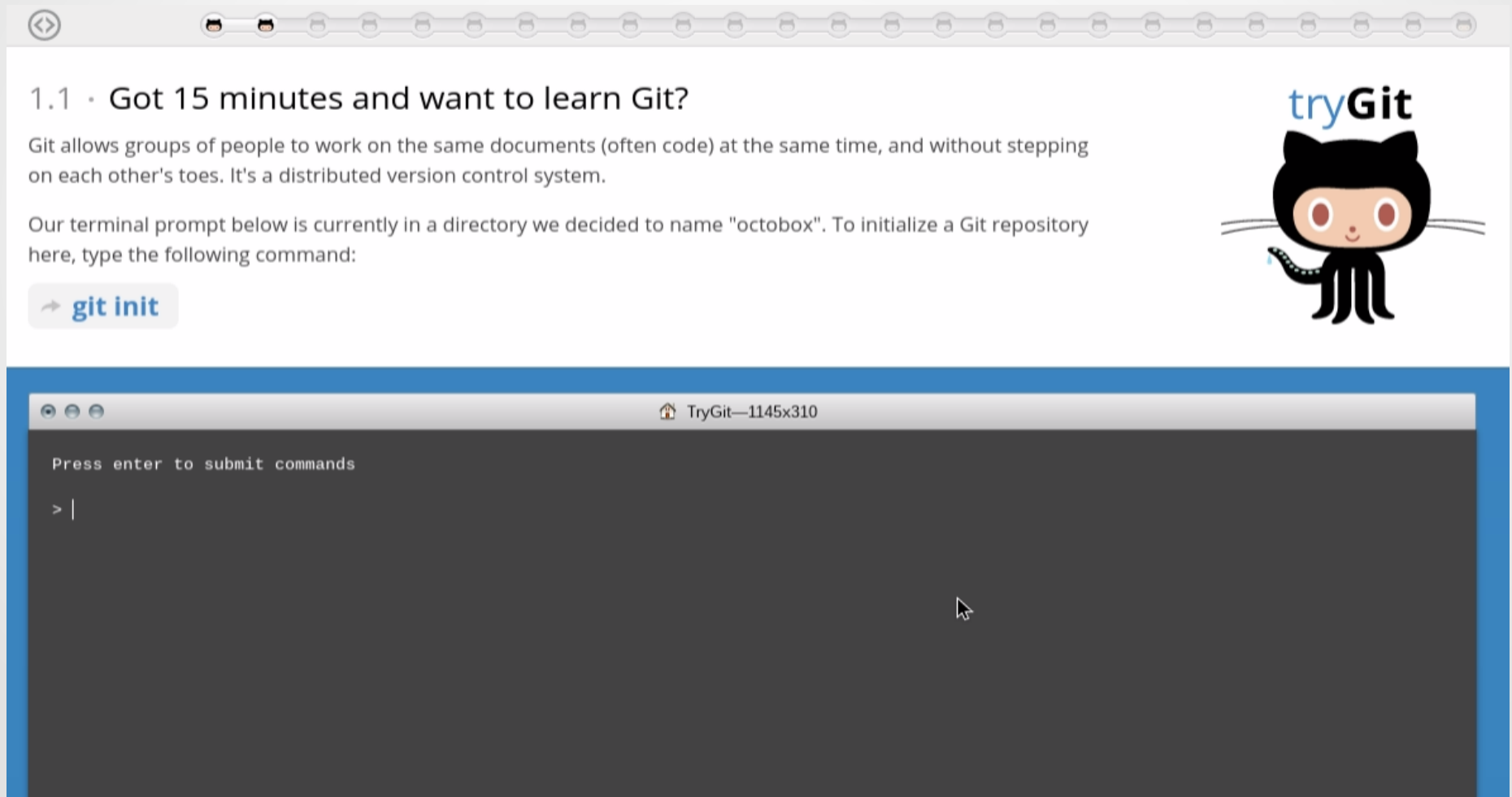
```
git init
```

para crear un nuevo repositorio de git.

hacer checkout a un repositorio

Tutoriales para iniciar

- <https://try.github.io>



The screenshot shows the tryGit website interface. At the top, there's a navigation bar with a back button and a series of GitHub Octocat icons. The main content area has a heading "1.1 · Got 15 minutes and want to learn Git?". Below this, a paragraph explains that Git allows groups of people to work on the same documents at the same time. Another paragraph states that the terminal prompt is currently in a directory named "octobox" and provides the command "git init" in a button. To the right of the text is the tryGit logo and the GitHub Octocat mascot. At the bottom, there's a terminal window titled "TryGit—1145x310" with a dark background. The terminal shows the prompt "Press enter to submit commands" and a cursor at the start of a new line.

1.1 · Got 15 minutes and want to learn Git?

Git allows groups of people to work on the same documents (often code) at the same time, and without stepping on each other's toes. It's a distributed version control system.

Our terminal prompt below is currently in a directory we decided to name "octobox". To initialize a Git repository here, type the following command:

➔ `git init`

tryGit

TryGit—1145x310

Press enter to submit commands

> |

Gui (Graphical User Interface)


- <https://git-scm.com/docs/git-gui>
 - git gui
 - gitk
 - Instalar: `dnf/apt-get install gitk git-gui`
- En windows:
 - También se instala todo esto.
 - Incluido git-bash
- + otras decenas.
- + integración en mucho IDE.

Github! Nuevo repositorio

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner

 pablodav ▾

Repository name

/ linuxpay ✓

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

Description (optional)

repositorio personal para compartir con linuxpay

☒  **Public**
Anyone can see this repository. You choose who can commit.

☐  **Private**
You choose who can see and commit to this repository.

☒ **Initialize this repository with a README**

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** ▾

Add a license: **MIT License** ▾ ⓘ

Create repository

Github! Linuxpay

- Este repo no es oficial, pero deja un ejemplo:
- <https://github.com/pablodav/linuxpay>

The screenshot shows the GitHub repository page for `pablodav / linuxpay`. The repository is described as a "repositorio personal para compartir con linuxpay". It has 2 commits, 1 branch, 0 releases, 1 contributor, and is licensed under MIT. The latest commit by pablodav is titled "agrego la presentacion" and was made a minute ago. The repository contains three files: `LICENSE`, `README.md`, and `presentacion_git.odp`. The `README.md` file is currently selected and displays the title "linuxpay" and the same description: "repositorio personal para compartir con linuxpay".

File	Commit Message	Time
<code>LICENSE</code>	Initial commit	13 minutes ago
<code>README.md</code>	Initial commit	13 minutes ago
<code>presentacion_git.odp</code>	agrego la presentacion	a minute ago

Llaves ssh

¿Para que una llave ssh?

- Podrás usar git clone/push/pull sin usar tu contraseña cada vez que lo precisas o sin tener que exponerla en el url.
- Generar una:
 - `ssh-keygen`
- Copiar a un servidor remoto:
 - `ssh-copy-id usuario@servidor`
- Copiar llave publica
 - `cat ../.ssh/id_rsa.pub`
 - Luego se puede poner en tu perfil del usuario github o gitlab

Extras de Github, Gitlab, otros

- Permiten repositorios a través de https.
 - Prefiero ssh, pero puede servir si se está detrás de un proxy.
- Permite manejo de issues + versiones, etc.
 - Las versiones son muy útiles para agrupar issues como en un hito/objetivo.
- Integran muy bien los tags de git para usar como release de un software (solo sirve para desarrollo pero es realmente útil, nota: se pueden usar tags de git sin estos servicios de igual forma).

Otros enlaces

- <https://github.com/jlord/git-it-electron>
- <https://services.github.com/resources/>
 - Muy recomendable git-cheatsheet esp.