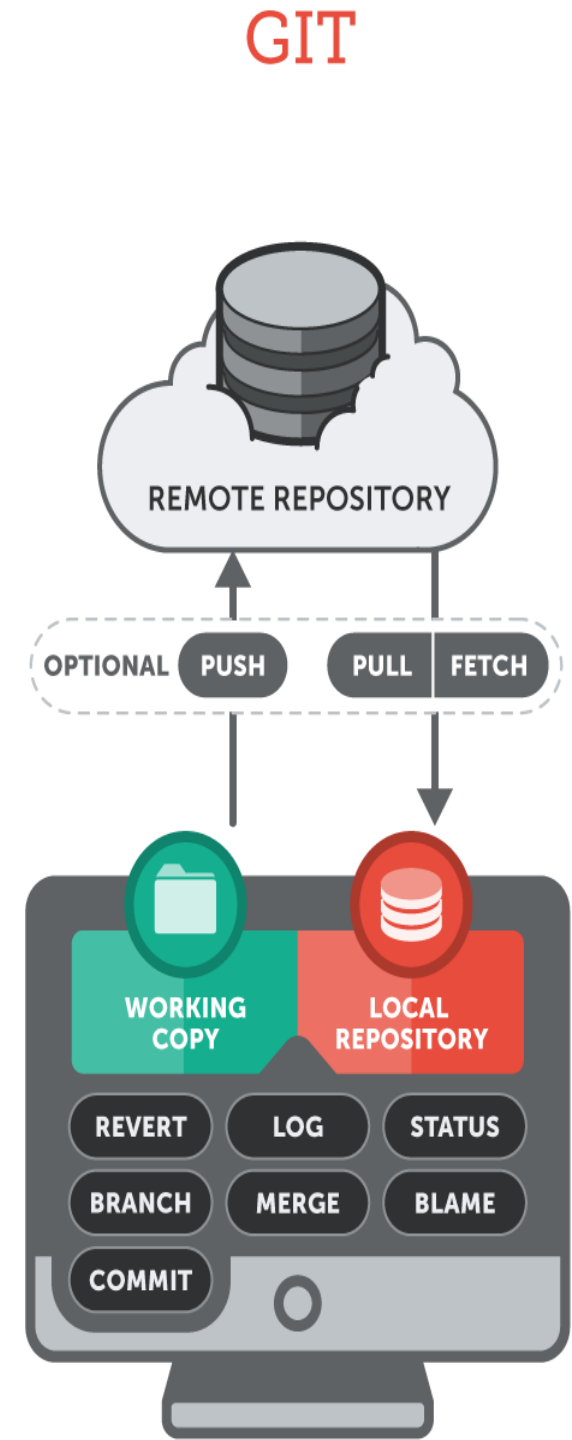
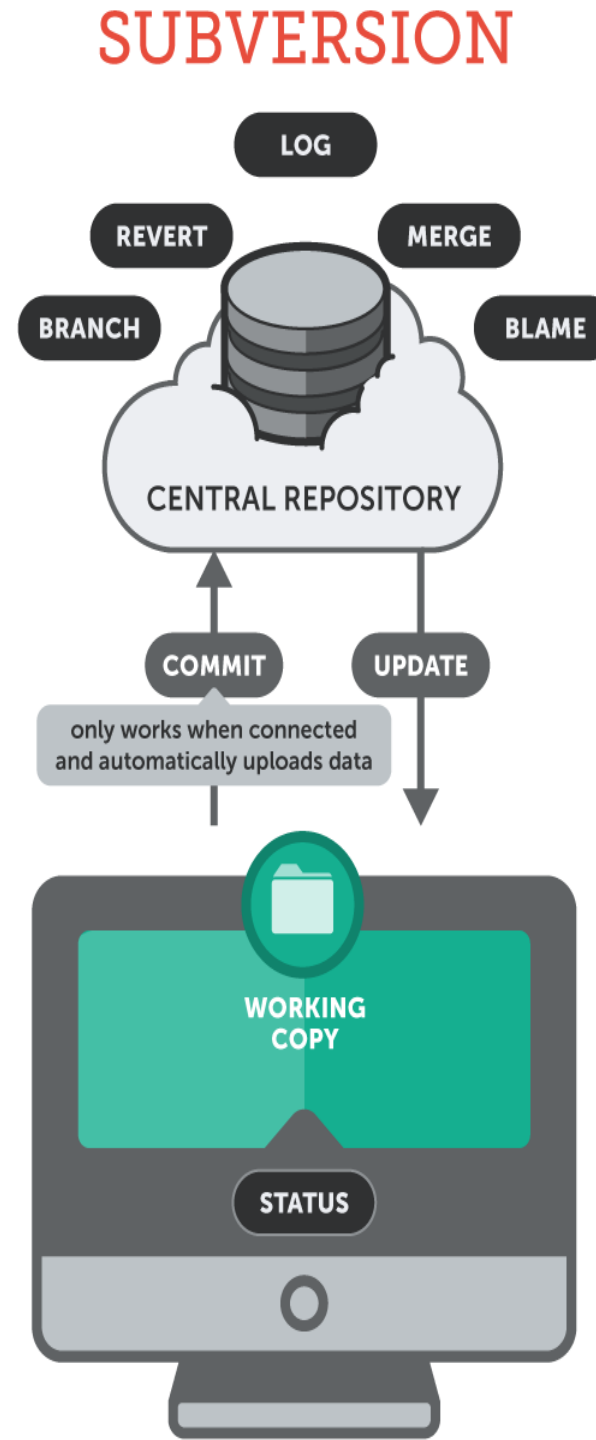


GIT

Git es un sistema de control de versiones distribuidas de código abierto y gratuito diseñado para manejar todo, desde proyectos pequeños a muy grandes, con velocidad y eficiencia.

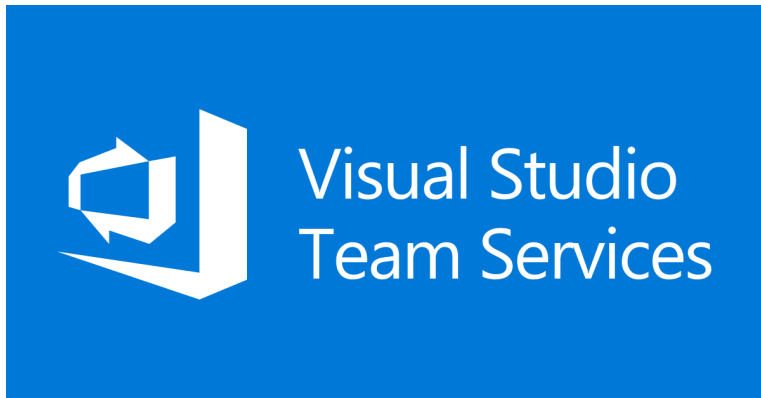
SUBVERSION

Al usar estos sistemas, todos los desarrolladores acceden a un único repositorio central. Si se realiza un cambio, llegará a todos los desarrolladores antes de que puedan comprometer sus cambios y, lamentablemente, eso también incluye un código roto.



Servicios de gestión de repositorios.

Hoy, los servicios de gestión de repositorios son componentes clave del desarrollo colaborativo de software. Permiten a los desarrolladores de software administrar los cambios en el código fuente y los archivos relacionados, crear y mantener múltiples versiones en un lugar central.



GitLab

COMANDOS DE GIT

- ▶ `git init` -> Crear un nuevo repositorio GIT
- ▶ `git config --global user.email "su correo de git"` -> Asigna su usuario al repositorio (`--local` para solo la carpeta)
- ▶ `git add [archive , (.), (-A)]`. -> Para agregar archivos a un stage (index)
- ▶ `git commit -m [Mensaje]` Agrega a una pila llamada Head lo que hay en mi stage
- ▶ `git push origin master` -> Envía los cambios a la rama master del repositorio remoto
- ▶ `git clone [url del repositorio]` -> Para clonar un repositorio
- ▶ `git remote add [url]` -> Conectar a un repositorio remoto

GUIA VISUAL

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

<https://code.visualstudio.com/docs/introvideos/versioncontrol>

Descargar Git

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

Tutorial: <https://launchschool.com/books/git/read/github>

The screenshot shows a web browser window displaying a GitHub repository page. The browser's address bar shows the URL `github.com/joesmith2444/my-first-repository`. The repository name `joesmith2444 / my-first-repository` is displayed at the top of the page. To the right of the repository name are buttons for `Unwatch` (with a dropdown arrow and the number 1) and `Star` (with the number 0).

The main content area is titled **Quick setup — if you've done this kind of thing before**. It contains several sections for setting up the repository:

- Quick setup — if you've done this kind of thing before**: This section offers two options for cloning the repository: `Set up in Desktop` (highlighted in green) or `HTTP` (with a dropdown arrow). The `SSH` option is also available. The URL `https://github.com/joesmith2444/my-first-repository.git` is provided for cloning. Below this, a note states: "We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#)."
- ...or create a new repository on the command line**: This section provides a list of commands to initialize a new repository and push it to GitHub:

```
touch README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/joesmith2444/my-first-repository.git
git push -u origin master
```
- ...or push an existing repository from the command line**: This section provides a list of commands to push an existing repository to GitHub:

```
git remote add origin https://github.com/joesmith2444/my-first-repository.git
git push -u origin master
```
- ...or import code from another repository**: This section states: "You can initialize this repository with code from a Subversion, Mercurial, or TFS project." and includes an `Import code` button.

On the right side of the page, there is a sidebar with a list of links: `<> Code`, `Issues` (with 0 issues), `Pull Requests` (with 0 pull requests), `Wiki`, `Pulse`, `Graphs`, and `Settings`.

SITIO GITHUB.IO

1

Create a repository

Head over to [GitHub](#) and [create a new repository](#) named *username.github.io*, where *username* is your username (or organization name) on GitHub.

If the first part of the repository doesn't exactly match your username, it won't work, so make sure to get it right.

Create a new repository

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

Owner / Repository name

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

Description (optional)

☐ 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: Add a license: ⓘ

Create repository

2

Clone the repository

Go to the folder where you want to store your project, and clone the new repository:

```
~ $ git clone https://github.com/username/username.github.io
```

3

Hello World

Enter the project folder and add an index.html file:

```
~ $ cd username.github.io  
~ $ echo "Hello World" > index.html
```

4

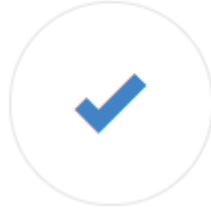
Push it

Add, commit, and push your changes:

```
~ $ git add --all  
~ $ git commit -m "Initial commit"  
~ $ git push -u origin master
```


...and you're done!

Fire up a browser and go to <https://username.github.io>.



► <https://pages.github.com/>