

git









&

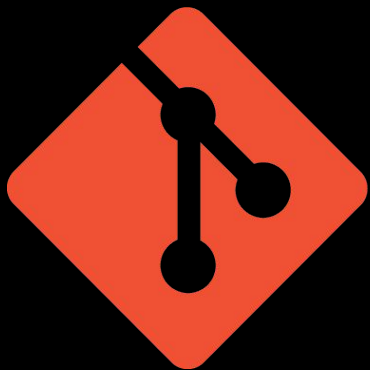


GitHub

...

Alvarado Morán Óscar Anuar
Jonathan Martiñón
Programación Concurrente
IIMAS, UNAM

- ▶  version beta
- ▶  version final
- ▶  version final (no es la final)
- ▶  version final (no mostrar al cliente)
- ▶  version final 2.0
- ▶  version final arreglos
- ▶  version final B
- ▶  version final final final esta sí es



git

¿Qué es?

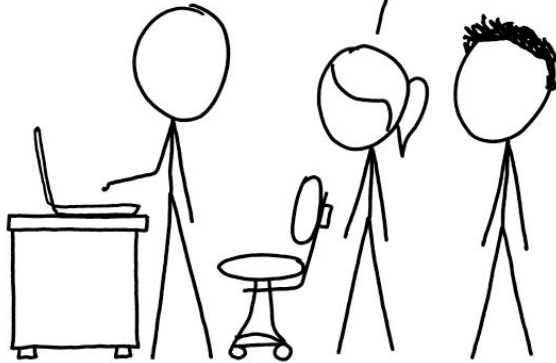
Sistema de Control de Versiones (VCS por sus siglas en inglés)

- Coordina versiones.
- Coordina trabajo entre múltiples desarrolladores.
- Quién y cuándo se hicieron cambios.
- Vuelve atrás en cualquier momento.
- Maneja repositorios locales y remotos.

THIS IS GIT. IT TRACKS COLLABORATIVE WORK
ON PROJECTS THROUGH A BEAUTIFUL
DISTRIBUTED GRAPH THEORY TREE MODEL.

COOL. HOW DO WE USE IT?

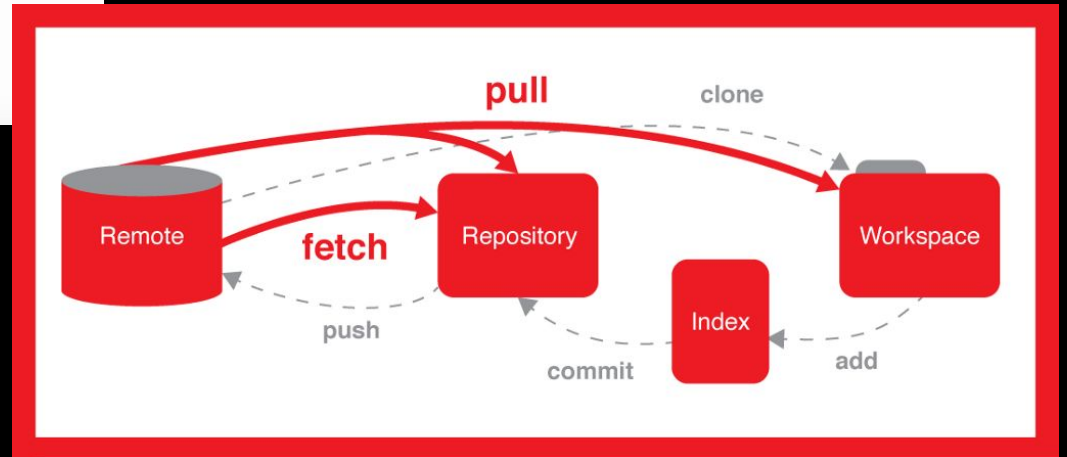
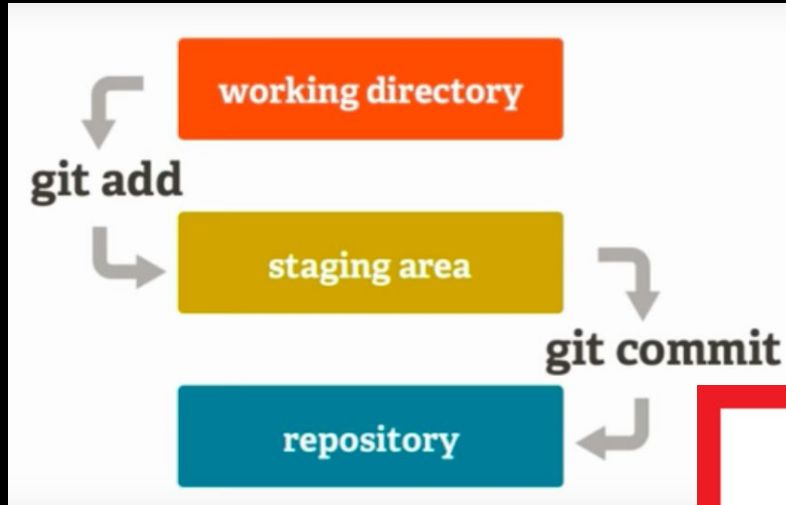
NO IDEA. JUST MEMORIZE THESE SHELL
COMMANDS AND TYPE THEM TO SYNC UP.
IF YOU GET ERRORS, SAVE YOUR WORK
ELSEWHERE, DELETE THE PROJECT,
AND DOWNLOAD A FRESH COPY.



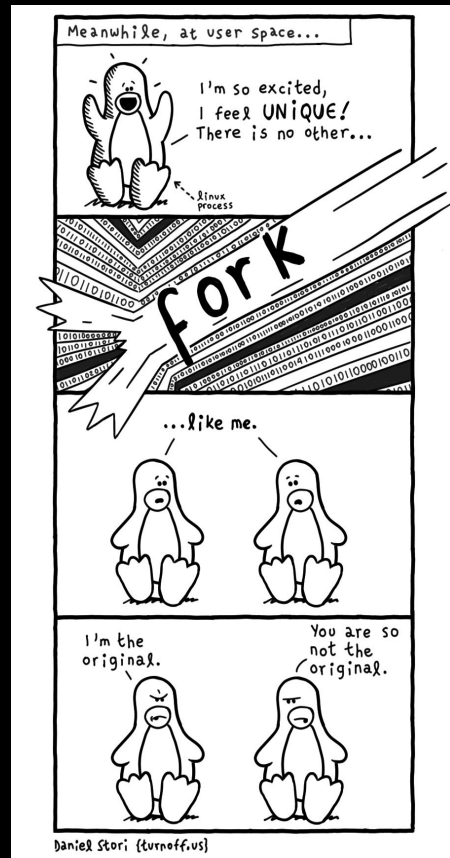
Comandos básicos:

- `git init`
- `git add`
- `git status`
- `git commit`
- `git push`
- `git pull`
- `git clone`

Estructura



Fork



¿Qué es?

- **Sitio web y servicio en la nube que ayuda a los desarrolladores a almacenar y administrar código**

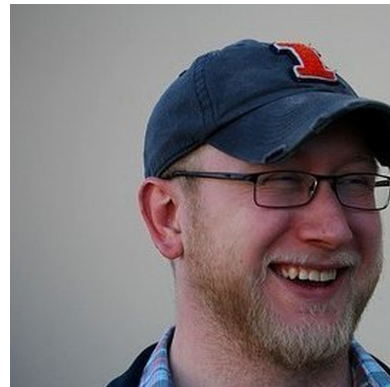


Orígenes

Desarrollado por:

- **Chris Wanstrath**
- **P. J. Hyett**
- **Tom Preston-Werner**
- **Scott Chacon**

usando Ruby on Rails, en 2008.



Comandos para crear un repositorio en Git desde la terminal de linux y luego subirlo a GitHub

- `git init` (desde la carpeta donde tenemos los archivos a subir)
- `git config --global user.mail "Tu_email@ejemplo.com"`
- `git config --global user.name "Tu Nombre"`
- `git add <archivo>` ó `git add .` ó `git add *.txt` (tipo de texto)
- `git status`
- `git commit <archivo>` ó `git commit -m "Comentario sobre el commit"`
- `git remote add origin <repositorio que en este caso sacaremos de GitHub (link)>`
- `git push -u origin master` (una rama en general)

Comandos para manipular archivos en un repositorio de git ya habiendo subido todo a GitHub

- `git add <archivo>` ó `git add .` ó `git add *.txt` (tipo de texto)
- `git status`
- `git commit <archivo>` ó `git commit -m "Comentario sobre el commit"`
- `git push`
- `git log`
- `git checkout -- <archivo>`
- `git branch <rama nueva>`
- `git checkout <rama a la que queremos desplazarnos>`
- `git merge <rama para fusionar con la actual>`
- `git checkout <código de un commit que quiera consultar>`
- `git pull <rama de la cual quieras jalar información a tu repositorio local>`

Otros comandos necesarios:

Los siguientes dos son para darse de alta como usuario:

- `git config --global user.mail "Tu_email@ejemplo.com"`
- `git config --global user.name "Tu Nombre"`

Este siguiente es para deshacer el último commit. Volver en el tiempo:

- `git checkout -- <archivo>`

Este comando es para ver los snapshots o fotos del repositorio. Todo lo que hacemos (y quién lo hace) después de un commit, pues:

- `git log`

Para checar las diferencias entre documentos cambiados:

- `git diff <archivo>`

Para hacer commit a todo lo del área de ensayo:

- `git commit -m "El commit de todos los archivos en la -staging area-"`

Para ver las ramas del proyecto:

- `git branch`

Otros comandos necesarios:

Para agregar nueva rama:

- `git branch <Nombre de nueva rama>`

Para ubicarse en otra rama

- `git checkout <Nombre de rama a la que se quiere entrar>`

Para agregar uno o varios archivos al área de ensayo:

- `git add .`
- `git add *.txt`

Para agregar dirección 'origen' para subir nuestro repositorio:

- `git remote add origin <dónde vamos a subir repositorio>`

Para subir por primera vez a nuestra rama:

- `git push -u origin rama`

Para agregar un archivo README.md al repositorio en GitHub:

- `echo "# Otro proyecto ni más ni menos" >> README.md`

Para que deje de ser un archivo .git

- `rm -rf .git`

Borrar de git sin borrarlo de los archivos locales:

- `git rm --cached archivo.org`

Otros comandos necesarios:

Para borrar ramas locales y en GitHub:

- `git branch -d rama`
- `git push origin --delete rama`

Para fusionar ramas:

- `git merge <rama para fusionar con la actual>`

Enlaces importantes

- [Descargar git desde la página.](#)
- [Tutorial para descargar git desde la terminal.](#)
- [Tutorial de git YouTube.](#)
- [Tutorial 2 de git YouTube.](#) (Este es el más completo)
- [Otro tutorial desde GitHub.](#)
- [Tutorial pull.](#)
- [Enlace para editar esta presentación y mejorarla haciendo pull request.](#)