

Intro a Git y Github

Diego Canul

Version Control System

Un VCS es un sistema que registra los cambios sobre un archivo o conjunto de archivos, de tal forma que se puede llamar a una versión en específico después.

Usar un VCS generalmente implica que si rompes tus archivos o los pierdes, fácilmente los puedes recuperar.

Git

Git es un *Distributed Version Control System* (DVCS), donde se accede a un repositorio que contiene las últimas versiones de los archivos así como su historial completo. Cada *clon* del repositorio contiene un respaldo completo de toda la información.

Git



<https://github.com/aspp-latam/2023-mexico-git-and-github>

Git

Verificar la configuración de git

```
git config --list
```

Si no hay una credenciales:

```
git config --global user.email "you@example.com"  
git config --global user.name "Your Name"
```

Mi primer repo

1. mkdir git_test
2. cd git_test
3. git init
4. git status
5. touch example.txt
6. git status
7. git add example.txt
8. git status
9. git commit -m "mensaje del commit"

Mi primer repo

```
(base) dacr@jupyter-eolica:~/git_test$ git commit -m "mi  
primer commit"  
  
*** Please tell me who you are.  
  
Run  
  
    git config --global user.email "you@example.com"  
    git config --global user.name "Your Name"  
  
to set your account's default identity.  
Omit --global to set the identity only in this repository.  
y.
```

Mi primer repo

1. `mkdir git_test`
2. `cd git_test`
3. `git init`
4. `git status`
5. `touch example.txt`
6. `git status`
7. `git add example.txt`
8. `git status`
9. `git commit -m "mensaje del commit"`
10. `git status`

El flujo de trabajo usual

1. `git status`
 2. Trabajo normal
 3. `git status`
 4. `git add file` ó `git commit -a -m "mensaje del commit"`
 5. `git commit -m "mensaje del commit"`
- ...

Dos comentarios

1. `git add .?`
2. La importancia del mensaje en el commit

Archivos no deseados

Crea un archivo y un directorio, agregalo a `.gitignore`

Regresar al pasado

Hacer cambios con precaución, en algunos casos los cambios son irreversibles.

Revisar el historial:

1. `git log`

No mantener los cambios en la versión actual es decir, desmodificar el archivo (modificar un archivo y revisar el status):

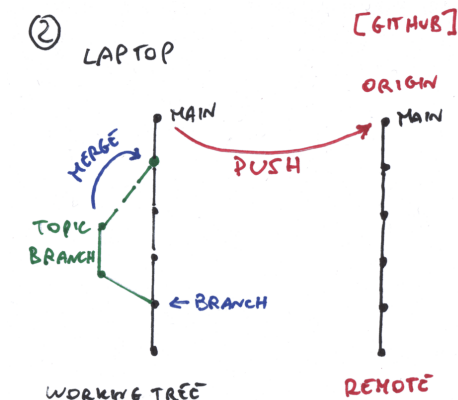
1. `git restore`

`git restore` es un **comando peligroso**, cualquier cambio hecho es ignorado. No debe ser usado a menos que estés seguro de no querer guardar los cambios hechos.

Github

Es una plataforma online que aloja Git y permite el desarrollo colaborativo

...



Configuración del SSH

En la terminal:

1. `ssh-keygen -t ed25519 -C "tu_correo_en_github@example.com"`
(enter, enter, enter)
2. `eval "$(ssh-agent -s)"`
3. `ssh-add ~/.ssh/id_ed25519`
4. `cat ~/.ssh/id_ed25519.pub` y copiar el output

En Github:

1. Ir a Settings
2. Claves SSH y GPG
3. Nueva clave SSH
4. Agregar clave

Comprobar la conexión en terminal:

1. `ssh -T git@github.com`
2. `:-)`

Clonando un repo

Otra forma de crear un repo es creando uno desde github.com y luego haciendo usando el comando `git clone github.com/tu_repo`.

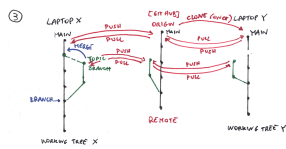
Puedes clonar un repo y usarlo de manera local o personal, o clonar un repo donde participas en un proyecto.

Interactuando con github I

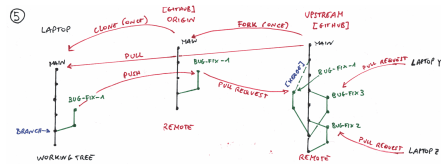
1. Abre el siguiente repo: https://github.com/dcanulr/git_eolica y copia el link clone ssh
2. Realiza el clon del repo `git clone git@github.com:dcanulr/git_eolica.git`
3. Crea un archivo nombrado "tu_nombre.csv" y actualiza el repo
4. Una vez hecho el commit local, revisa `git status` actualiza el repo remoto con `git push`

Interactuando con github II

1. Si existen cambios hechos en el repo remoto se verán reflejados en el status
2. Para actualizar los cambios remotos utiliza `git pull`
3. Trabaja de manera local y realiza el commit [local], y después realiza el push [remoto]



<https://github.com/aspp-latam/2023-mexico-git-and-github>



<https://github.com/aspp-latam/2023-mexico-git-and-github>