

Git & GitHub



Índice Sesión 2

- 1. Git**
- 2. GitHub**
 - Definición
- 3. Línea de Comandos**
 - Comandos Git

Git

Git es un gestor de **repositorios** de versiones software

- Desarrollado por Linus Torvalds en 2005 en código libre.
- Lo diseñó para soportar el desarrollo de Linux con un modelo descentralizado y abierto.

Git es un programa para hacer seguimiento a los cambios a través del tiempo, conocido en programación como **control de versiones**.

git es un comando de UNIX/Linux

- Documentación:

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

<https://guides.github.com/introduction/git-handbook/>

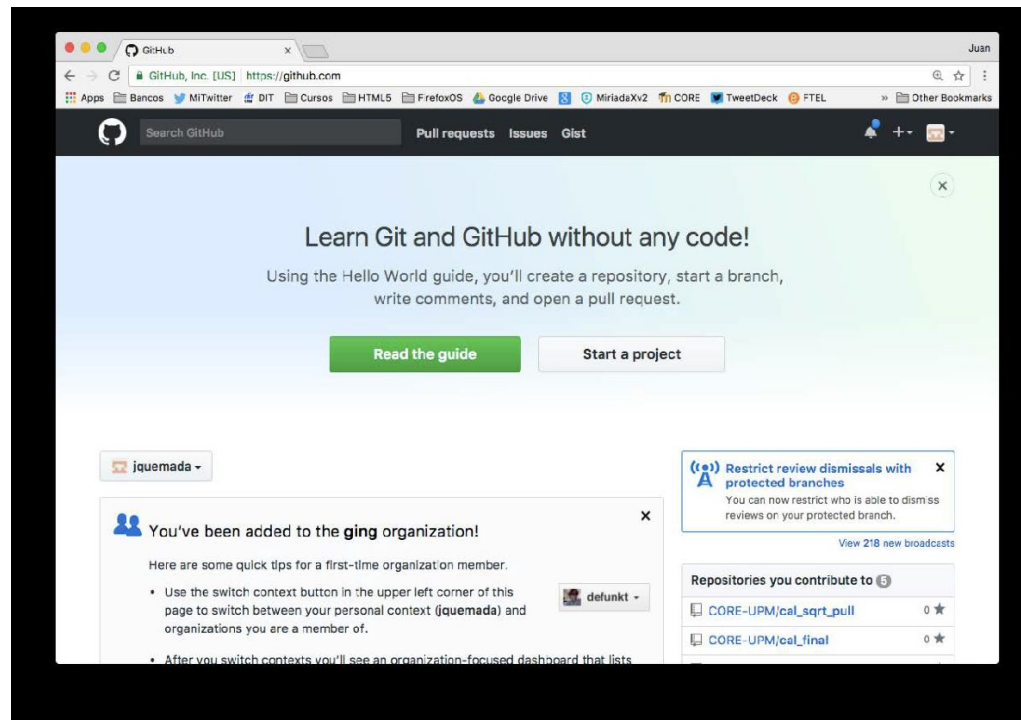
<https://services.github.com/on-demand/resources/cheatsheets/>

GitHub



- Portal donde programadores comparten repositorios con proyectos Git
- Nos da acceso a ellos a través del navegador Web y a través de Git

Este curso requiere tener cuenta en GitHub: <https://github.com>



GitHub



GitHub -> lema "Social coding"

- Red social donde programadores comparten repositorios remotos Git

Repositorios **públicos son gratis**, los privados de pago

El repositorio se identifica en un repositorio local con un **URL**, p. e.

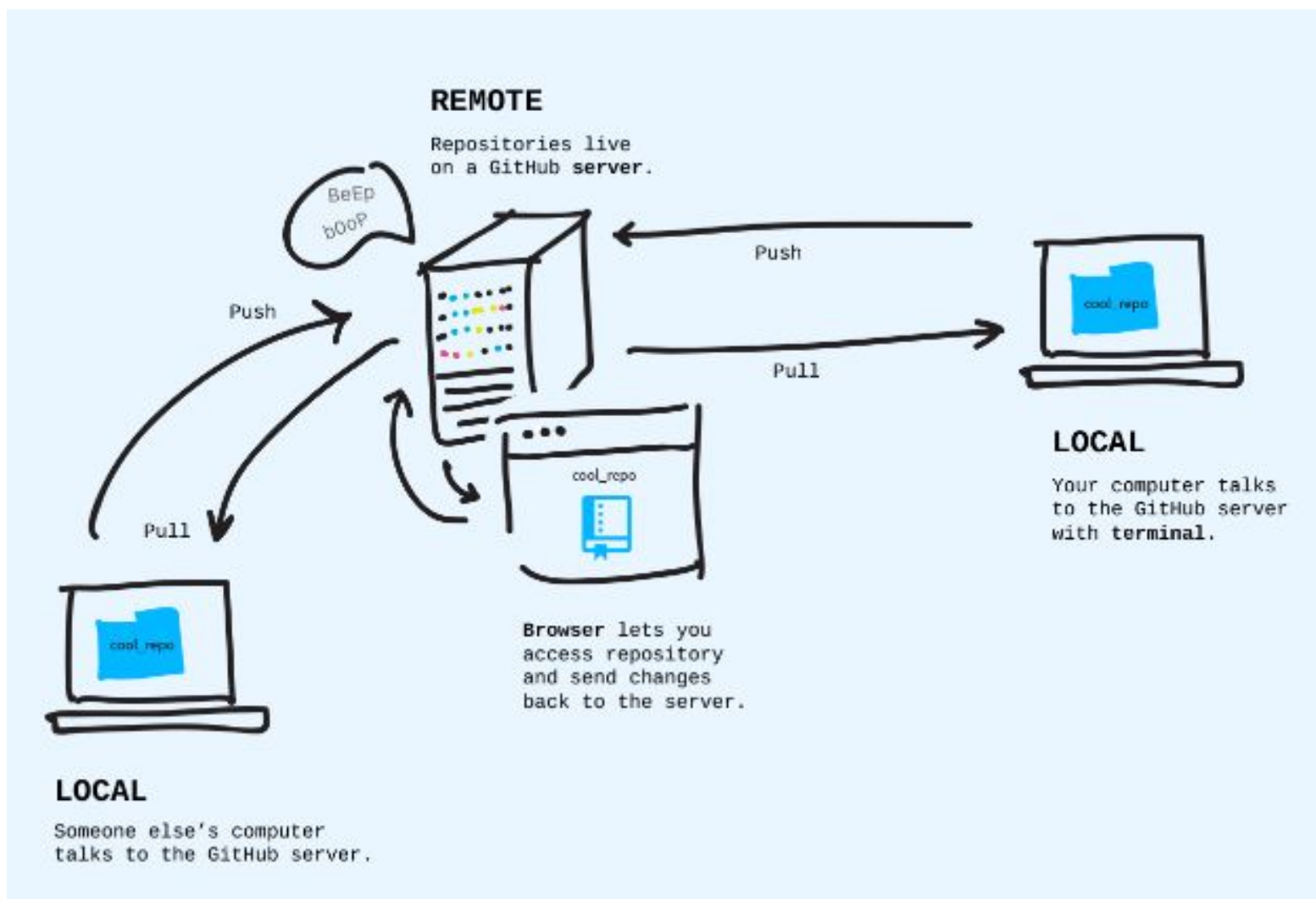
- <https://github.com/franguerrerosanchez/releevant> URL del repo en GitHub
- <https://github.com/franguerrerosanchez/releevant.git> explicita (equivalente)

GitHub es un sitio web que permite a personas de cualquier lugar subir el código que tienen en Git para que cualquier otra persona pueda fácilmente trabajar en conjunto en el mismo proyecto. GitHub actúa como el repositorio central para que tú y otras personas compartan. Tú empujas (push) tus cambios a Github y puedes obtener (pull) los cambios que han hecho otras personas.

Cuando pones algo en GitHub, esa copia vive en uno de los servidores de GitHub. Esto lo convierte en un repositorio **remoto (remote)** porque no está en tu ordenador, está en un servidor "remoto" en algún otro lugar. **Empujando (pushing)** tus cambios **locales** (los de tu ordenador) a éste, lo mantienes al día.

Otras personas siempre pueden obtener lo más reciente de tu proyecto **tirando (pulling)** de tus cambios desde el remoto (a otro ordenador). Así es cómo varias personas pueden trabajar juntas en un proyecto sin necesidad de tener acceso a tu ordenador, donde está almacenada tu copia local.

GitHub



GitHub

- Ve a github.com, haz login, y haz clic en el '+' en la parte derecha arriba y después en 'Nuevo repositorio' ('New repository').
- Dale un nombre que coincida con el nombre de tu repositorio local, 'hola-mundo', y una descripción corta.
- Hazlo público. Esto significa que será listado en tu perfil público.



Git: La línea de comandos



Existen muchas formas de usar Git.

Por un lado tenemos las herramientas originales de línea de comandos, y por otro lado tenemos una gran variedad de interfaces de usuario con distintas capacidades.

La línea de comandos es el único lugar en donde puedes ejecutar **todos** los comandos de Git - la mayoría de interfaces gráficas de usuario solo implementan una parte de las características de Git por motivos de simplicidad.

Si tú sabes cómo realizar algo desde la línea de comandos, seguramente serás capaz de averiguar cómo hacer lo mismo desde una interfaz gráfica. Sin embargo, la relación opuesta no es necesariamente cierta. Así mismo, la decisión de qué cliente gráfico utilizar depende totalmente de tu gusto, pero *todos* los usuarios tendrán las herramientas de línea de comandos instaladas y disponibles.

Instalación



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

<https://atom.io>

Comandos Git



git --version

- ¿Ya tienes instalado Git o no lo sabes? Escribe el comando en tu terminal y si devuelve un número de versión, entonces está instalado.

Comandos Git



git config --list

- si quieres comprobar tu configuración para mostrar todas las propiedades.

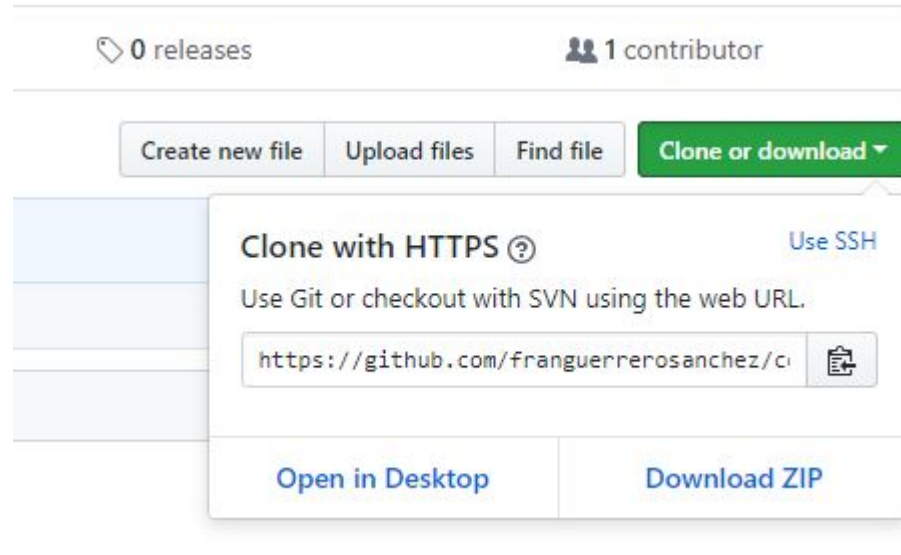
```
$ git config --list
user.name=John Doe
user.email=johndoe@example.com
color.status=auto
color.branch=auto
color.interactive=auto
color.diff=auto
...
```

Comandos Git



git clone url

- obtiene una copia de un repositorio existente.

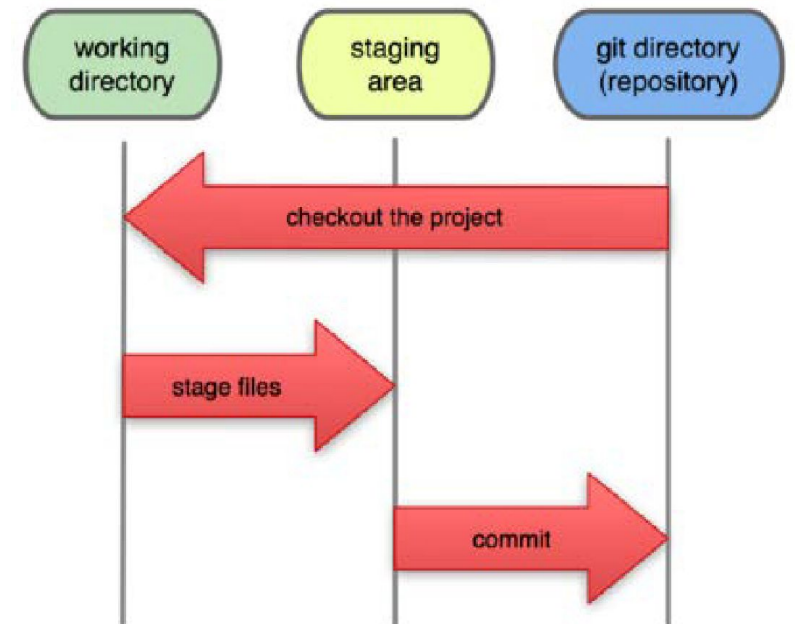


Comandos Git



git status

- muestra estado de los ficheros del directorio:
 - **modified**: modificados respecto al commit anterior
 - **untracked**: no existentes en el commit anterior
 - **staged**: registrados para el próximo commit

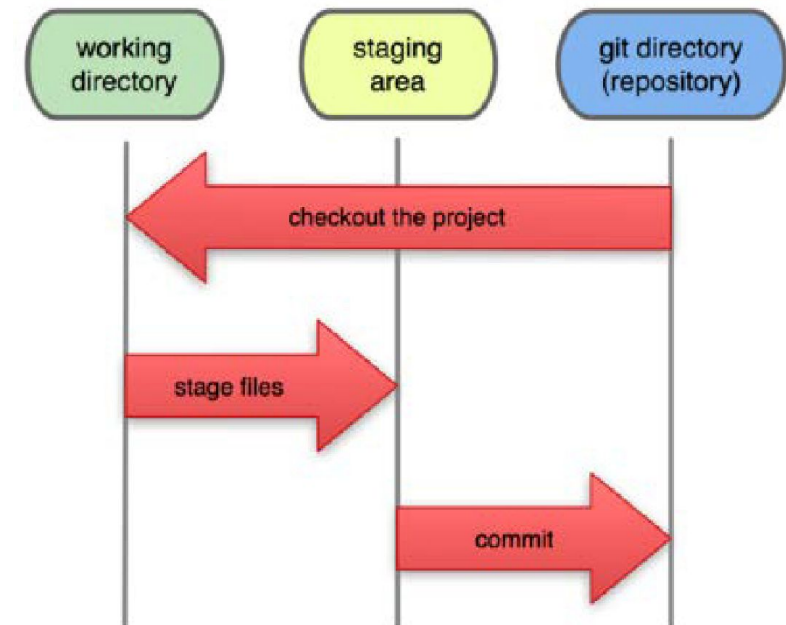


Comandos Git



git add ...

- registra en el índice los ficheros indicados
 - **git add .** registra en el índice todos los ficheros **nuevos** o **modificados**

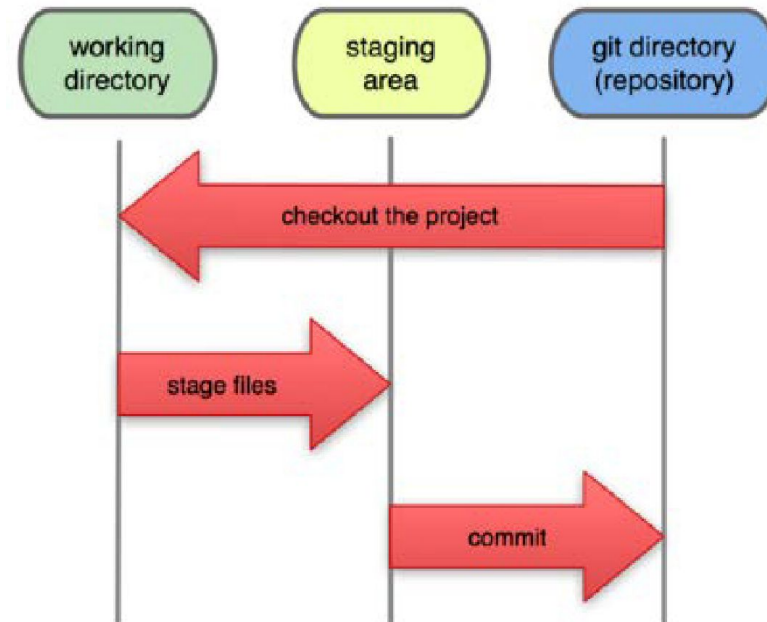


Comandos Git



git commit ...

- Genera un nuevo commit con lo registrado en el índice
- **git commit -m "Descripción"** guarda nuevo commit con mensaje o título "Descripción"



Comandos Git



git push

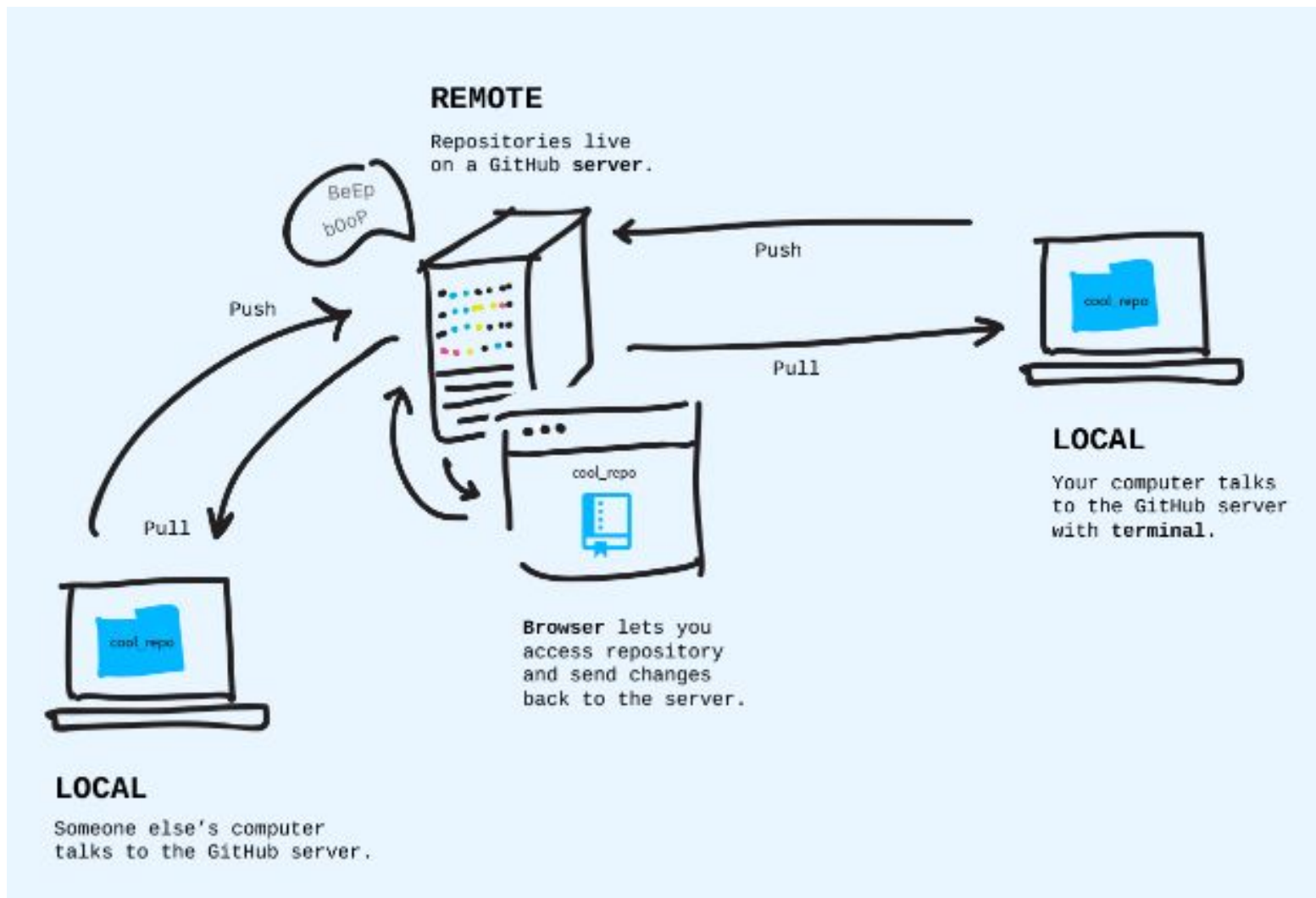
- Lo siguiente es hacer **push** (empujar) todo lo que has hecho localmente a tu repositorio remoto en GitHub. Esto es algo que harás con frecuencia para que tu versión remota esté al día y coincida con el estado tu versión local.

git pull



1. Clonar [repositorio compañero](#)
2. Crear un atom un fichero “nombre_apellido” que contenga la frase “Hola Git”.
3. Subir los cambios al repositorio remoto.
4. Actualizar con “Pull” los cambios de los demás compañeros en el repositorio local.

GitHub



Comandos Git



git help

git branch

git pull

...

Referencias

- <https://git-scm.com/book/es/v2/Fundamentos-de-Git-Guardando-cambios-en-el-Repositorio>
- <https://github.com/jlord/git-it-electron>
- <http://try.github.io>