



Sistema de control de versiones Git (con GitHub)

Facundo Goyena - 2019



Sistema de control de versiones

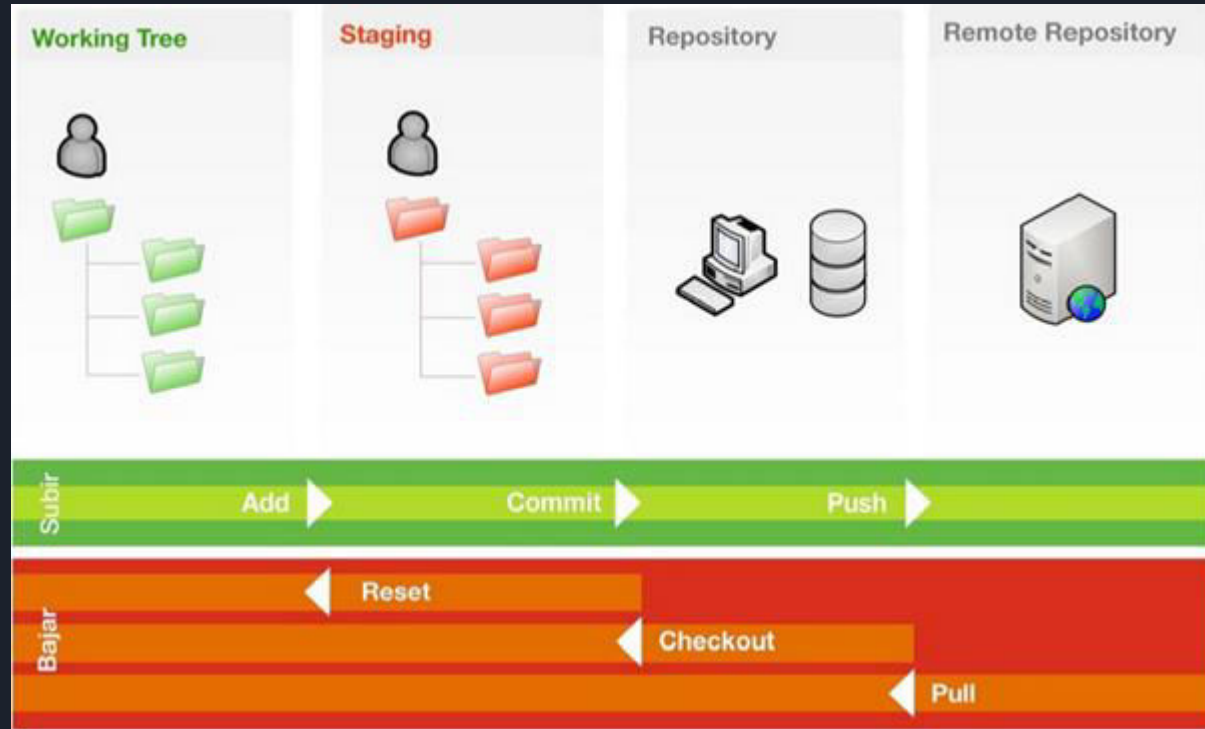
- Un sistema que mantiene registros de los cambios
- Permite el desarrollo colaborativo
- Permite saber quién y cuándo realizó cada cambio
- Permite revertir cualquier cambio



Git

- Sistema de control de versiones distribuido
 - Cada persona tiene un repositorio local
- No depende de tener internet
- Sistema de ramas (branches)

Git: Proceso de subida y bajada





Primeros Pasos

- Instalación
 - <http://git-scm.com/download/win>
- Crear cuenta en GitHub
 - www.github.com
- Configuración
 - `git config --global user.name "Facundo Goyena"`
 - `git config --global user.email facundo.goyena@gmail.com`
- Documentación
 - <https://git-scm.com/book/es/v1/>



Comenzando

- Inicializar un repositorio local

```
git init
```

- Agregar repositorio remoto

```
git remote add origin https://github.com/facundogoyena/prueba.git
```

- Clonar un repositorio

```
git clone https://github.com/facundogoyena/prueba.git
```



Creando *commits*

- Cambios específicos

```
git add README.txt  
git commit -m "Agregado de README"
```

- Todos los cambios

```
git add --all  
git commit -m "Todooooooooo"
```



Dónde estoy?

- Cambios pendientes, estado actual

```
git status
```

- Listado de *commits*

```
git log
```




Deshaciendo

- Quitar del *stage*

```
git reset HEAD archivo.js  
ó  
git rm --cached archivo.js
```

- Restaurar archivo

```
git checkout archivo.js  
ó  
git checkout [REV] -- archivo.js
```

Siendo [REV] el commit, tag, rama/branch

- Revertir commit

```
git revert 2654a9
```



Deshaciendo

- Quitar del *stage*

```
git reset HEAD archivo.js  
ó  
git rm --cached archivo.js
```

- Restaurar archivo

```
git checkout archivo.js  
ó  
git checkout [REV] -- archivo.js
```

Siendo [REV] el commit, tag, rama/branch

- Revertir commit

```
git revert 2654a9
```



Identificar cambios

- Cambios locales
`git diff`
- Cambios en *stage*
`git diff --cached`
- Cambios de una revisión
`git diff 2654a9`



.gitignore

- Excluye archivos de ser trackeados/agregados
- Ejemplo:

```
/build/  
/dist/  
*.exe  
  
.DS_STORE
```



Empujando al remoto

- Empuja los cambios actuales para la rama/branch dada
`git push <remote> <branch>`



Obteniendo del remoto

- Obtiene los cambios del repositorio remoto

```
git pull <remote> <branch>
```



Branches / Ramas

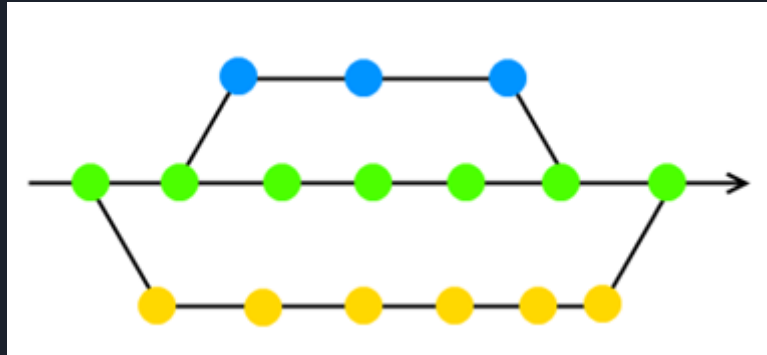
- Permite crear un contexto nuevo, basado en el branch anterior
- Establece un paradigma de cambios por branches
- Branch por defecto: **master**
- Crear un branch
`git checkout -b agregar_propiedades`
- Cambiar al branch
`git checkout agregar_propiedades`

Mover cambios de un branch a otro

- Merge

```
git checkout master
```

```
git merge agregar_propiedades
```



Basado en

<http://mgaitan.github.io/intro-git/index.html>

